VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ
FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY
INSTITUTE OF INFORMATICS

NÁVRH, TVORBA A IMPLEMENTACE SOFTWAROVÉ APLIKACE VE FIREMNÍM PROSTŘEDÍ

DESIGN, CREATION AND IMPLEMENTATION OF SOFTWARE APPLICATIONS IN THE CORPORATE ENVIRONMENT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — **Bc. František Brothánek**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Lukáš Novák, Ph.D.**
SUPERVISOR

BRNO 2021

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA PODNIKATELSKÁ

Zadání diplomové práce

Ústav:	Ústav informatiky
Student:	**Bc. František Brothánek**
Studijní program:	Systémové inženýrství a informatika
Studijní obor:	Informační management
Vedoucí práce:	**Ing. Lukáš Novák, Ph.D.**
Akademický rok:	2020/21

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

Návrh, tvorba a implementace softwarové aplikace ve firemním prostředí

Charakteristika problematiky úkolu:

Úvod
Vymezení problému a cíle práce
Teoretická východiska práce
Analýza problému a současné situace
Vlastní návrhy řešení, přínos návrhů řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Cílem práce je analyzovat, navrhnout a implementovat webovou aplikaci do firemního prostředí.

Základní literární prameny:

GÁLA, L., J. POUR a Z. ŠEDIVÁ. Podniková informatika. 2. přeprac. a aktualiz. vyd. Praha: Grada Publishing, 2009. ISBN 978-80-247-2615-1.

HARDCASTLE, E. Business Information Systems. Ventus Publishing ApS, 2008. ISBN 978-87-76-1-463-2.

PRETTYMAN, S. Learn PHP 7: object oriented modular programming using HTML5, CSS3, Javascript, XML, JSON, and MYSQL. Apress, 2015. ISBN 978-1-4842-1730-6.

SODOMKA, P. a H. KLČOVÁ. Informační systémy v podnikové praxi. 2. vyd. Brno: Computer

Press, 2000. ISBN 978-80-251-2878-7.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2020/21

V Brně dne 28.2.2021

L. S.

Mgr. Veronika Novotná, Ph.D.
ředitel

doc. Ing. Vojtěch Bartoš, Ph.D.
děkan

Abstrakt

Diplomová práce se zabývá návrhem, tvorbou a implementací webové aplikace pro online hlasování ve společnosti IS4U. Práce je rozdělená na tři části. V první části práce jsou popsány teoretické základy a východiska využité pro praktickou realizaci práce. Druhá část analyzuje současný stav firmy, jejího informačního systému a zákazníků a mapuje kritéria pro vytvoření webové aplikace. Poslední část se zabývá samotnou realizací nové webové aplikace a zhodnocením dosažených výsledků. Nová webová aplikace zjednodušuje a zabezpečuje hlasování uživatelů.

Klíčová slova

webová aplikace, univerzitní informační systém, volby, hlasování, volební lístek

Abstract

The main goal of the diploma thesis is to design, create, and implement the web application for online voting in the company IS4U. The thesis is divided into three parts. The first part describes the theoretical foundations required for the practical implementation. The second part analyzes the current state of the company, its information system, customers, and criteria for creating a web application. The last part deals with the implementation of the new web application and evaluates the achieved results. The web application simplifies and secures voting by its users.

Keywords

web application, university information system, elections, voting, ballot

Bibliografická citace

BROTHÁNEK, František. Návrh, tvorba a implementace softwarové aplikace ve firemním prostředí [online]. Brno, 2021 [cit. 2021-05-14]. Dostupné z: https://www.vutbr.cz/studenti/zav-prace/detail/133704. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Lukáš Novák.

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 16. května 2021

……

podpis autora

Poděkování

Mé poděkování patří Ing. Lukášovi Novákovi Ph.D. za odborné vedení a rady, které tvarovaly tuto diplomovou práci. Také chci poděkovat vedení společnosti IS4U za poskytnuté materiály a příležitost zpracovat vybraný projekt pod jejich záštitou. Poslední poděkování, ne však nejmenší, patří mé milované manželce Lence za podporu a motivování, bez kterého bych sotva dokončil tuto práci.

OBSAH

Úvod

Pandemie koronaviru ještě více zdůraznila potřebu převádění procesů vyžadujících lidské setkávání do online prostoru. Společnost si přivyká na řadu každodenních úkonů, které nově provádí elektronicky.

Tyto změny probíhají i v akademickém prostředí. Zkoušky se skládají pomocí e-learningu z domova, přednášky a cvičení probíhají pomocí videokonference. Pod vlivem těchto okolností je jinak nahlíženo i na proces hlasování. Vysoké školy a další instituce používají princip hlasování na řešení množství rozhodovacích problémů – ať už to jsou rozpočty nebo je to výsledná známka ze státní závěrečné zkoušky.

Z výše popsaných důvodů se v diplomové práci zabývám návrhem, tvorbou a implementací webové aplikace umožňující online hlasování. V první části práce se zaměřím na podstatné teoretické pojmy, které povedou k hlubšímu porozumění následujících kapitol. Druhá část zanalyzuje firmu, pod jejíž záštitou bude webová aplikace vyvíjena, bude provedena analýza zákazníků, jež produkt požadují, a představím všechna klíčová kritéria, co aplikace musí splňovat. Třetí, stěžejní část, předloží rizikovou a časovou analýzu, návrh databázového schématu, samotnou aplikaci s jejími částmi a zakončí shrnutím finančního a nefinančního zhodnocení práce.

VYMEZENÍ PROBLÉMU A CÍLE PRÁCE

Hlavním cílem diplomové práce je provést analýzu, vytvořit návrh a implementovat webovou aplikaci do prostředí firmy. Aplikace slouží k tajnému hlasování uživatelů informačního systému. Práce je zaměřená na analýzu současného stavu hlasování bez využití informačního systému, návrh nejen finančně efektivního řešení, ale zejména řešení splňující požadavky zákazníků a vedení firmy. Celá aplikace bude umístněná do informačního systému společnosti, proto musí být se systémem plně kompatibilní a navázaná na jeho data.

Z toho plynou dílčí cíle práce, jako je analýza trhu s již existujícími řešeními, analýza procesu, časová analýza a celková formulace požadavků a kritérií na aplikaci. Na konci se práce zabývá implementací aplikace do informačního systému.

1 TEORETICKÁ VÝCHODISKA PRÁCE

Teoretická část diplomové práce povede k pochopení základních pojmů a vazeb mezi nimi. Vzhledem k tomu, že se práce zabývá návrhem, tvorbou a implementací softwarové aplikace ve firemním prostředí, zvolil jsem tři základní pojmy, o které se práce opírá: data a informace, systém a informační systém. V dalších kapitolách se pak budu zabývat také pojmy databáze, kryptografie, webové aplikace, volby a hlasování.

1.1 ZÁKLADNÍ POJMY

Níže jsou nastíněny základní pojmy, které pomohou k pochopení problematiky zpracované v dalších kapitolách.

1.1.1 DATA A INFORMACE

S pojmem data či informace se setkáváme snad každý den. Nejjednodušeji by se data dala charakterizovat jako něco, co předchází informacím. Informace pak je již hodnota, kterou z dat získáváme. Informace vzniká z dat v okamžiku jejich použití. [1]

N. Wiener, jeden ze zakladatelů kybernetiky, informaci formuloval následovně: „Informace je informací, není to ani hmota, ani energie. Materialismus, který toto nepřipouští, nemůže přetrvat dnešek.“ [2]

Existují však i jiné definice, které jsou pro potřeby této práce vhodnější: „Informace je zpráva o nestálém jevu, která u nás (příjemců) snižuje míru neznalosti o tomto jevu.“ [3] anebo „data, kterým jejich uživatel přisuzuje určitý význam a které uspokojí konkrétní objektivní informační potřebu svého příjemce.“ Nositelem této informace je například obraz, zvuk, text, či schéma. [1] Pokud bychom však pojem informace vztáhli k informačním systémům, nejčastějším nosičem informace je číselná, či textová hodnota.

1.1.2 SYSTÉM

Dalším základním pojmem je systém, který navazuje na pojem data a informace. „Systém je účelově definovaná neprázdná množina prvků a množina vazeb mezi nimi, přičemž vlastnosti prvků a vazeb mezi nimi určují vlastnosti (chování) celku.“ [3]

Systém je také definován jako soubor komponent, které spolupracují na dosažení společného cíle. Úkolem systému je přijímat vstupy a transformovat je na výstupy. [4]

Pro každý systém rozlišujeme a definujeme následující vlastnosti:

- **Účel systému**, respektive cílové chování systému
- **Strukturu systému**, tj. prvky systému a vazby mezi nimi
- **Vlastnosti prvků** systému významné pro celkové chování systému
- **Vlastnosti vazeb** mezi prvky systému, významné pro celkové chování systému
- **Okolí systému**, tj. vymezení prvků, které již nepatří do systému, ale jejichž vlastnosti a vazby systému na tyto prvky okolí významným způsobem ovlivňují chování systému
- **Případné subsystémy**, které znamenají rozdělení systému na menší relativně samostatné a uzavřené celky uvnitř systému, pokud je to možné. [3]

Vyjmenované vlastnosti musíme brát v potaz při návrhu jakéhokoliv informačního systému.

1.1.3 INFORMAČNÍ SYSTÉM

Pojem systém ještě rozšířím o informační systém. „Informační systém je soubor lidí, technických prostředků a programů, zabezpečujících sběr, přenos, zpracování a uchování dat, za účelem prezentace informací pro potřeby uživatelů činných v systémech řízení.“ [5]

Naproti tomu informační technologie jsou určité znalosti, metody a nástroje, jak zpracovávat data a získávat z nich informace. Vztah mezi informačním systémem a informačními technologiemi můžeme znázornit: informační systém reprezentuje touhu po informacích, informační technologie jako takové je naplňují. [1]

1.2 DATABÁZE

Databáze jsou v informačních systémech místa, ve kterých můžeme uchovávat, zaznamenávat a číst obraz skutečnosti uložený v podobě dat.

1.2.1 RELAČNÍ DATABÁZE

Při návrhu datového schématu v druhé kapitole práce využívám relační model. Z toho důvodu se zabývám definicí tohoto databázového modelu.

Jedním z nejpoužívanějších datových modelů v současnosti patří (vedle objektového modelu) relační model.

Pomocí tohoto systému jsme schopni zaznamenat nejenom data o zkoumaných objektech, ale i vzájemné vztahy těchto objektů, a tudíž je nám umožněno přiblížit se pomocí tohoto systému blíže k reálnému světu. [6]

Data jsou uložena ve formě tabulek ve sloupcích a řádcích. Data mohou být propojena pomocí cizích klíčů. [7] Tabulky mají svá specifika:

1. Každý řádek odpovídá n-tici relace (jednomu záznamu)
2. Pořadí řádků je nevýznamné
3. Žádné dva řádky nejsou stejné (neobsahují duplicity)
4. Význam každého sloupce je určen názvem atributu
5. Žádné dva atributy nejsou stejné
6. Hodnoty ve sloupcích jsou atomické

1.2.2 INTEGRITA RELAČNÍHO MODELU

Při modelování dat se setkáváme s určitými omezeními plynoucími z reálného světa. Integritu modelu chápeme jako stav, při kterém data uložená v modelu odpovídají vlastnostem reálného světa. Rozlišujeme několik typů integritních omezení.

1. Integritní omezení pro entity
    a. Doménová integrita – znamená, že hodnoty musí pocházet z určité domény – tzn. splňuje specifikaci povolených hodnot daného atributu.
    b. Každá relace musí mít Primární klíč.[1]

[1] Primární klíč je jednoznačná (unikátní) a minimální (atomická) nenulová hodnota, na kterou je následně v relacích odkazováno.

c. Referenční integrita – Cizí klíč je takový atribut, který se odkazuje na primární klíč jiné n-tice – hodnota musí být stejná, jako odkazovaný primární klíč. Databáze nesmí obsahovat cizí klíč na neexistující primární klíč.

2. Integritní omezení pro vztahy

Vztahy mohou nabývat 3 různé typy kardinality: 1 : 1, 1 : N (taktéž N:1), N:M. Tento poměr uvádí, kolik n-tic relací si sobě navzájem odpovídá.

Příklad poměru 1:1 může být „člověk“ a „řidičský průkaz.“ Jeden člověk vlastní jeden řidičský průkaz a zároveň jeden řidičský průkaz je ve vlastnictví jednoho člověka.

1:N je například diplomová práce a vedoucí práce. Jedna diplomová práce má jednoho vedoucího práce a zároveň jeden vedoucí práce vede N (neboli více) diplomových prací.

M:N může být například student a předmět. 1 student studuje N předmětů a zároveň 1 předmět je studován více studenty. [6].

Správně navržený relační model tudíž splňuje požadavky pro vytvoření jednotné báze dat, která poskytuje pravdivá a spolehlivá data pro všechny uživatele, splňuje tzv. „jednu verzi pravdy“. [8]

1.2.3 JAZYK SQL

Publikovaný článek *„A Relational Model Of Data For Large Shared Data Banks“* od autora Dr. Ted Codd v roce 1970 byl návrhem implementace nového datového modelu, který byl nazván relačním. Ten se stal základem pro rozšíření relačních databází, jak je známe dnes. Pomocí základních operací relační teorie (sjednocení, kartézský součin, rozdíl, selekce, projekce a propojení) lze uskutečnit veškeré operace s daty. Ostatní operace jsou již jen vzájemné kombinace výše uvedených. Všechny tyto operace měl být schopen provést nový dotazovací jazyk SQL (Structured Query Language), který vznikl v průběhu 80. let.

Jazyk byl postupně přijat jako standard různými výrobci databázových aplikací a stal se také spojovacím článkem, protokolem, mezi různými systémy a aplikacemi. Pomocí konceptu klient/server se dotazy definují na straně klienta, pošlou se na stranu serveru, který dotaz vyhodnotí, realizuje a výsledek pošle jako odpověď klientovi.

Jazyk SQL se skládá z několika částí:

1. **DDL** – Data Definition Language; pomocí tohoto jazyka jsou definovány struktury dat
2. **SDL** – Storage Definition Language; definuje způsob ukládání tabulek
3. **VDL** – View Definition Language; pomocí tohoto jazyka lze ukládat pohledy
4. **DML** – Data Manipulation Language; pro koncové uživatele nejpoužívanější, definuje základní příkazy – Insert, Update, Delete a nejpoužívanější příkaz Select. [6]

Zabývám se tímto jazykem z toho důvodu, že je součástí systému, ve kterém je vytvořena webová aplikace týkající se předkládané diplomové práce.

1.3 KRYPTOGRAFIE

Cílem mé práce je navrhnout, vytvořit a implementovat volební aplikaci jako součást informačního systému ve firemním prostředí, kde hlasování v této aplikaci má být ověřitelné, ale zároveň tajné. Z toho důvodu je třeba použít metody, kterými se zabývá vědní disciplína kryptografie.

Nutnost použití utajené komunikace je patrně stejně stará, jako samo písmo a posílání zpráv, protože již od pradávna existují zprávy se zvlášť závažným obsahem. U těchto zpráv hrozilo, že pokud by se dostaly do nesprávných rukou, napáchaly by významnou škodu. Sice se způsob a forma (médium) předávání zpráv změnilo, potřeba utajení zpráv je možná ještě důležitější než v dávných dobách. [9]

Kryptografie je nauka o metodách utajování zpráv v prostředí, kde existuje třetí strana – jiní potenciální příjemci než cíloví.

Šifra je algoritmus, podle kterého je pevně definovanými kroky možné zašifrovat a dešifrovat zprávu. Šifra ve většině případů vyžaduje klíč. Klíč je pomocná informace, pomocí které je možné provést šifrování a dešifrování. [10]

Komunikační protokol (neboli jasný předpis procesu komunikace) musí v moderní formě komunikace splňovat následující kritéria.

- **Důvěrnost** – utajení obsahu zprávy

- **Integrita** – bez možnosti změnit zprávu
- **Autentizace** – nikdo cizí se nemůže vydávat za odesílatele
- **Nepopiratelnost** – lze ověřit vše, co všichni účastníci komunikace učinili

Mezi nejznámější způsoby, jak narušit tuto komunikaci patří:

- **Zabránění doručení –** útočník znemožní komunikaci
- **Vložení jiné zprávy –** útočník se vydává za odesílatele
- **Odposlech** – útočník odposlouchává mezi komunikačními partnery
- **Předstírání jiné identity –** nazýváno také anglicky jako *„Man in the middle“* – útočník odposlouchává, ale i předstírá, že je protistrana a oproti odposlechu může měnit zprávu.

Vyjmenovanými kritérii se věnuji detailněji níže.

1.3.1 AUTENTIZACE

Autentizace je proces ověření identity subjektu neboli ověření, že daný subjekt je opravdu tím, za koho se vydává. Subjektem je každý uživatel systému. Metoda prokázání je založena na prokázání identity. Existují 3 druhy ověření:

- **Autentizace dle toho, co subjekt zná** – subjekt zná heslo (nejčastěji PIN, heslo, kontrolní otázka)
- **Autentizace dle toho, co subjekt má** – subjekt něco vlastní (platební karta, čip, klíč, mobilní telefon s generátorem kódu pro dvojfázové ověřování, například Google Authenticator)
- **Autentizace dle toho, co subjekt je** – subjekt se prokáže biometrickými údaji (otisk prstu, snímání dlaně, duhovky, obličeje)

Proto rozlišujeme 3 typy autentizace:

- **Jednofázová autentizace** – prokázání se jednou ze tří typů autentizace
- **Dvoufázová autentizace** – prokázání se dvěma ze tří typů autentizace
- **Třífázová autentizace** – prokázání se všemi třemi typy autentizace. [11]

1.3.2 AUTORIZACE

Tento proces zahrnuje získávání souhlasu s provedením určité operace (zápis a čtení dat). Například v informačním systému vysoké školy nesmí mít student přístup k editaci známek ze zkoušek, čtení známek cizích studentů či osobních údajů jiných studentů. Naproti tomu učitel musí mít přístup k editaci známek ním vyučovaných předmětů. [12]

1.3.3 SYMETRICKÉ ŠIFROVÁNÍ

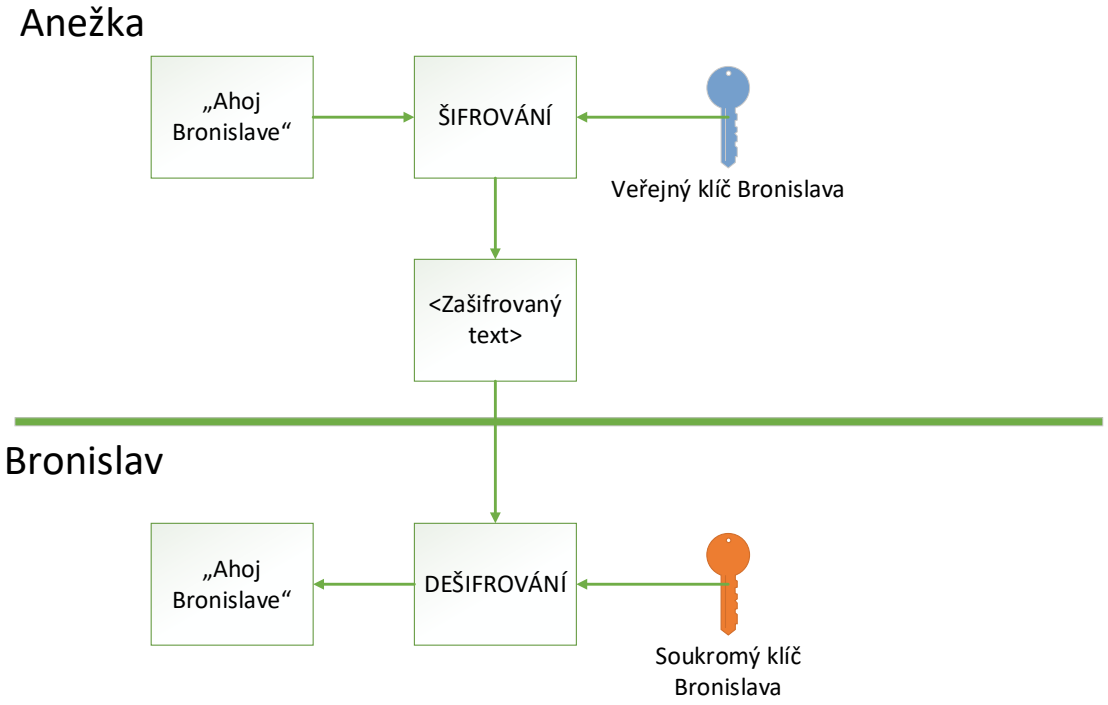
Symetrické šifry používají stejný šifrovací klíč pro šifrování i dešifrování. Šifry bývají zpravidla relativně rychlé, co se výpočetního výkonu týče, a při použití dostatečně velkého klíče (standartně až 256 bitů) bezpečné. Nevýhodou je ale problematika domluvy na společném klíči po nezašifrované lince. Při odposlechu se může šifra vyzradit a komunikace tím pádem může být kompromitovaná (může dojít k vyzrazení, podvrhu zprávy a předstírání jiné identity). [12]

1.3.4 ASYMETRICKÉ ŠIFROVÁNÍ

Asymetrické šifrování funguje na principu šifrování soukromými klíči. Tato metoda je založena na jistém matematickém problému (např. prvočíselný rozklad násobku dvou velkých prvočísel, kalkulace diskrétního logaritmu apod.). Každý komunikační partner si vygeneruje pár klíčů – veřejný a soukromý. Veřejný klíč je všem dostupný a kdokoliv jej může použít pro zašifrování zprávy příjemci. Privátní klíč není sdílený a je pro dešifrování zprávy zašifrované právě pomocí veřejného klíče ze stejného páru. Postup komunikace je následující (pro komunikační subjekty jsem zvolil jména Anežka a Bronislav):

1. Anežka si vygeneruje pár klíčů, soukromý nikomu neposkytne, veřejný zveřejní. Bronislav si také (nezávisle na Anežce) vygeneruje pár klíčů, soukromý nikomu neposkytne, veřejný zveřejní.
2. Anežka chce poslat zprávu Bronislavovi, tudíž zprávu zašifruje pomocí Bronislavova veřejného klíče a pošle ji Bronislavovi. Jelikož zprávy zašifrované veřejným klíčem jde dešifrovat jen privátním klíčem ze stejného páru, Bronislav si dešifruje zprávu a přečte.

3. Bronislav pošle odpověď Anežce tak, že zašifruje zprávu veřejným klíčem Anežky a pošle jí ji. Ta je jako jediná schopna zprávu dešifrovat svým privátním klíčem.



Obrázek 1: Ukázka komunikace s asymetrickým šifrováním

(Zdroj: vlastní zpracování)

Mezi nejznámější asymetrické šifry patří RSA, Diffie-Hellman a DSA. [12]

1.3.5 HASHOVACÍ FUNKCE (HASH)

Pro podporu integrity dat byla poptávka po funkci, která by mohla zajistit ověření integrity. Tato funkce však neslouží pouze k tomuto účelu, pro mou diplomovou práci je však definice dostačující. Tato funkce se nazývá hashovací. Jedná se o algoritmus, který splňuje následující vlastnosti:

- převede vstupní data do relativně malého čísla s daným počtem cifer
- je těžké najít dva vstupy se stejným výsledkem hashovací funkce
- je těžké najít opačnou funkci

Používané hashovací funkce jsou funkce z rodiny SHA (SHA-1, SHA-2, SHA-3). [12]

Pro demonstraci slouží následující tabulka použití hashovací funkce SHA-3-256. V prvním případě byl vstup „Ahoj Bronislave“, v druhém případě byla zaznamenána gramatická chyba ve jméně.

Tabulka 1: Demonstrace Hashovací funkce

(Zdroj: vlastní zpracování)

Vstup	Výstup
Ahoj Bronislave	5f712c36aad4064cf1f380fc8ec8067cdc8967a36dcacb8d8d225f91
Ahoj Cronislave	e7de41064133890c06be80ab47b8955f7c2ea67f7cabb064cff473fa

Hashovací funkce je vhodným nástrojem pro to, aby například hlasování jednotlivce nebylo podvrhnuté. Z toho důvodu jsem se rozhodl tuto funkci využít při návrhu vlastního řešení.

1.3.6 ELEKTRONICKÝ PODPIS

Jedná se o mechanismus, kterým se zjišťuje, zda dokument je pravý – splňuje podmínku integrity a autentizace. Elektronický podpis se vytváří ve dvou krocích.

1. Spočítá se hash dokumentu
2. Výsledný hash se zašifruje soukromým klíčem uživatele, který podpis vytváří. Soukromým klíčem šifrovaný hash se nazývá elektronickým podpisem.

Verifikace dokumentu probíhá v následujících krocích

1. Příjemce sám vytvoří hash z dokumentu
2. Příjemce dešifruje přijatý elektronický podpis veřejným klíčem odesílatele.
3. Příjemce porovná hash s dešifrovaným elektronickým podpisem. Pokud se rovnají, dokument nebyl změněn a dokument byl vytvořen jedině majitelem soukromého klíče odesílatele.

Elektronický podpis dokazuje, že ten, kdo podepsal dokument, vlastní soukromý klíč. Na rozdíl od šifrování se používá veřejný klíč k dešifrování a soukromý klíč k šifrování. [13]

1.3.7 ČASOVÁ RAZÍTKA

Pomocí elektronického podpisu jsme schopni prokázat autorství a integritu dokumentu, nejsme ale schopni prokázat rozhodný čas, ke kterému byl soubor vytvořen. Systémový čas je možné jednoduše podvrhnout, proto je potřeba, aby existovala všeobecně uznávaná autorita, která je schopna tento čas nezměnitelným způsobem dokumentu přidružit. Toto řeší tzv. časová razítka (anglicky „time stamps“) a tzv. DV-certifikáty, což jsou potvrzení o držení či pravosti podpisů dokumentu k danému času.

Proces požádání o časové razítko je jednoduchý.

1. Klient spočítá hash dokumentu a tento hash vloží do žádosti o časové razítko.
2. Autorita vydávající časová razítka, vytvoří časové razítko – elektronicky podepsaná datová struktura, kterou nelze podvrhnout. [13].

1.4 DIAGRAMY A MATICE

Diagramy slouží k názornému objasnění procesů, matematických vztahů či myšlenek. Diagramů, matic a podobných pomůcek využívaných při návrhu softwaru je mnoho. Ty, které budou využity v této práci, budou představeny v následujících podkapitolách.

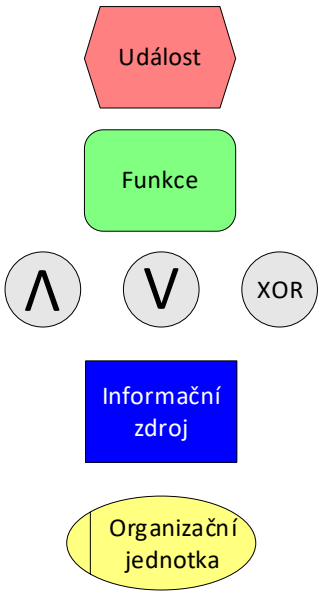
1.4.1 EPC DIAGRAM

Metoda EPC (Event-driven Process Chain) se řadí v oblasti návrhu a vývoje softwaru k nejrozšířenějším, protože byla zakomponována do mnoha systémů, jako SAP či ARIS. Metoda spočívá v řetězení událostí a aktivit do řetězce realizující požadovaný cíl. Obecně je diagram EPC vstupní podmínkou pro cílový stav procesu. V návrhu EPC diagramu se využívá následujících elementů:

1. Aktivity (Activities) – jsou základním stavebním blokem a určují, co je v rámci procesu vykonáno.
2. Události (Events) – jedná se o situace, které náleží před, či za aktivitou. Jedná se o stavy procesu.
3. Logické spojky (Connectors) – spojují jednotlivé aktivity a události. Logické spojky mohou slučovat buď více vstupů, nebo více výstupů. Spojky mohou nabývat logických hodnot:

a. ∧ AND – „a zároveň“
b. ∨ OR – „nebo“
c. XOR – „vzájemně se vylučující nebo“

Mimo standardní EPC diagram je dnes využíván tzv. eEPC (extended EPC), který umožňuje doplnit do diagramu i další doplňkové prvky: Informační zdroj, který aktivita využívá, či jaká Organizační jednotka vykonává danou aktivitu. [14]



Obrázek 2: EPC Diagram – vysvětlivky

(Zdroj: vlastní zpracování)

1.4.2 MATICE ODPOVĚDNOSTI

Matice odpovědnosti popisuje kompetence a odpovědnost jednotlivých organizačních jednotek, či rolí. Role definované v RACI matici mohou být zachyceny i v EPC diagramu. RACI je akronym pro následující role:

R – Responsible – subjekt vykonává daný úkol.

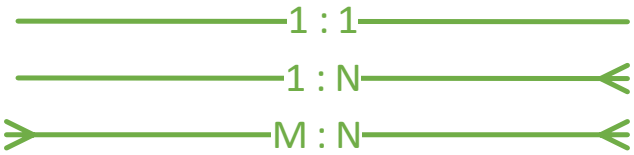
A – Accountable – subjekt je odpovědný za provedení práce.

C – Consulted – se subjektem je řešení konzultováno.

I – Informed – subjekt je o zpracování průběžně informován. [15]

1.4.3 ENTITO-RELAČNÍ DIAGRAM

Pro návrh databáze využíváme zápis pomocí entito-relačního modelu, který nám pomůže zachytit všechny klíčové části návrhu databáze. Entito-relační model se dá zakreslit více způsoby – v této práci je zachycena metoda ERD ve zjednodušeném stylu zápisu. Ve schématu jsou rozlišeny vazby 1:1, 1:N a N:M.



Obrázek 3: ERD diagram, význam vazeb

(Zdroj: vlastní zpracování)

Mimo vazby ER diagram obsahuje tabulky, které vazby spojují. [6]

1.5 WEBOVÉ APLIKACE

Návrh mé aplikace je součástí webového informačního systému. Architektura webových aplikací se nejvíc blíží centralizovanému výpočetnímu modelu s mnoha distribuovanými „tenkými“ klienty, které se připojují k „tlustému“ serveru, kde se zpracovává převážná část dat. Webové aplikace využívají jazyka HTML a jako primární transportní médium je využit protokol HTTP. Tyto technologie se staly populární díky rozšíření sítě World Wide Web.

1.5.1 HTML

HTML (z angličtiny „Hypertext Markup Language“) je programovací jazyk pro vyváření webových dokumentů určených k zobrazování na internetu. Podstatou jazyka jsou jednotlivé elementy, tzv. „tagy“. Příponu má „html“, či „htm“. Pro interpretaci HTML souboru se využívá webový prohlížeč. [16]

1.5.2 CSS

CSS (z angličtiny „Cascading Style Sheets“, neboli „Kaskádové styly“) je jazyk sloužící k popisu stylu zobrazení elementů na webových stránkách HTML, nebo XML. CSS

popisuje rozložení stránky, barvu a font textu a další charakteristiky webové stránky. Hlavním smyslem tohoto jazyka je umožnit návrhářům oddělit vzhled dokumentu od obsahu. V jazyce HTML se jednotlivý obsah připojí k stylům, a tudíž webový prohlížeč bude vědět, jak graficky vykreslit webovou stránku. [17]

1.5.3 JAVASCRIPT

JavaScript je programovací jazyk, jehož počátky sahají do roku 1993. Jedná se o objektově orientovaný skriptovací jazyk, který je řízen událostmi. Je známý a využívaný především ve webových aplikacích, kde poskytuje dynamicky se měnící obsah bez nutnosti opakovaného načtení stránky. [18] Pomocí tohoto jazyka je možné reagovat na aktuální stav formuláře stránky. Je možné skrývat či zobrazovat prvky formuláře, jako jsou například checkboxy nebo tlačítka závislá na interakci uživatele, a rovněž průběžně stahovat data z externího serveru.

1.5.4 PERL

Perl je skriptovací programovací jazyk vyvinutý v roce 1987 lingvistou, programátorem Larry Wallem. Díky rozvoji internetu se Perl stal velmi oblíbeným jazykem pro tvorbu skriptů webových aplikací. Největšího rozšíření Perl dosáhl ve verzi 4 roku 1991. Verze 5 přinesla podporu výkonnějších datových struktur a objektů. Silné stránky jazyka spočívají zejména v kvalitní podpoře regulárních výrazů a rychlém a pohodlném zpracování textových řetězců. [19] Slabé stránky jsou zejména ve zmatenosti jazyku a zastaralosti (existují vhodnější, mladší alternativy jako například PHP, Python nebo ještě modernější některé Javascript frameworky jako Angular, či React). [20]

Tvorba webových aplikací pomocí Perlu spočívá ve využívání knihoven CGI (Common Gateway Interface). Tyto knihovny jsou komunikačním protokolem mezi webovým serverem a externí aplikací. CGI je nástroj, díky němuž se mohou vypsat informace z databáze u klienta, řešit politiku uživatelských účtů apod. [21]

1.5.5 SSL/ TLS

Protokol SSL vytvořila firma Netscape. Oficiálním protokolem internetu se však stal protokol TLS (Transport Layer Security protocol) vycházející z protokolu SSL verze 3.

Protokol SSL verze 3 a protokol TLS jsou si velice blízké, avšak klient TLS se „nedomluví“ se serverem SSL a opačně. Jak klient, tak i server musí být naprogramován na protokol TLS, nebo SSL. [13]

1.5.6 HTTP

HTTP (z angličtiny „Hypertext Transfer Protocol“) je protokol určený pro komunikaci s webovými servery. Protokol slouží pro přenos souborů HTML, ale i XML[2] apod. Typicky používá port TCP 80. [13] V současné době je možné ho použít pro přenos jakéhokoliv souboru (podobně jako příloha e-mailu).

Uživatel (obvykle pomocí webového prohlížeče) zašle dotaz serveru obsahující cestu k souboru a informaci o webovém prohlížeči, jazyku, operačním systému apod. Server poté odpoví stavovým kódem, zda se mu podařilo najít soubor, jestli k němu má uživatel přístup a poté (pokud je vše v pořádku) soubor zašle. Prohlížeč u uživatele vykreslí soubor do grafické podoby v prohlížeči a čeká na to, až uživatel vytvoří další dotaz.

1.5.7 HTTPS

Pro aplikační protokol HTTP provozovaný přes SSL/TLS používáme URI schéma HTTPS. Jedná se o dvě vrstvy – první, SSL/TLS, provádí autentizaci za využití certifikátů, druhá, protokol http, provádí základní autentizaci, například jménem a heslem. [13]

1.6 ČASOVÁ ANALÝZA

Časová analýza se zabývá procesy pro řízení projektu z pohledu času. Při provádění časové analýzy je důležité dodržet logické návaznosti a optimální termíny zahájení a ukončení projektu.

Časová analýza obecně sestává z následujících činností:

[2] XML je značkovací jazyk, který umožňuje vytváření konkrétních aplikací s účelem otevřeného přenosu různých typů dat.

Definování činností – rozplánování jednotlivých činností projektu

Seřazení činností – činnosti na sebe musí navzájem navazovat, aby bylo možné sestavit harmonogram činností.

Odhad délky trvání činností – neboli doby potřebné pro realizaci činnosti (pro projekty, které jsou plánované, se používá expertní odhad a zpětná analýza podobných projektů; pro hodnocení již uplynulého projektu jsou zdroje ze záznamů).

Sestavení časového rozvrhu – harmonogram na základě doby a zdrojů.

Sestavení kritické cesty – využívají analýzy CPM, PERT a další… [22]

Metodu CPM a PERT blíže rozvedu v následující kapitole, jelikož je metoda PERT využita i prakticky v návrhové části práce.

1.6.1 CPM A PERT

Metoda CPM (z angličtiny „Critical Path Method“) je algoritmus pro plánování činností projektu. Využívá se ke zpětnému ohodnocení projektu.

Metoda PERT (z angličtiny „Program Evaluation and Review Technique“) je zobecněná metoda CPM. PERT je výhodnější z důvodu vyšší přesnosti odhadů – doba trvání není totiž přesně daná, ale počítá se jako vážený průměr ze tří odhadů – optimistický termín (a), pesimistický termín (b) a pravděpodobný termín (m).

Z těchto tří odhadů je vypočítán vážený průměr podle následujícího vzorce:

$$t = \frac{a + 4m + b}{6}$$

Jako výstup metody CPM, nebo PERT dostáváme termíny jednotlivých činností a informaci, zda mají časovou rezervu, případně jak velkou. Termíny bez časové rezervy leží na tzv. „kritické cestě“. Termíny, které leží na kritické cestě, se snažíme dodržet, protože prodloužení termínu činnosti ležící na kritické cestě zapříčiní prodloužení celé délky trvání projektu.

Pomocí časové analýzy také získáváme harmonogram projektu – kalendářní data pro řízení projektu. [22]

1.7 ANALÝZA RIZIK

Riziko je nebezpečí vzniku škody, poškození, ztráty či nezdaru při podnikání. V projektovém řízení se riziko definuje jako určitá pravděpodobnost vzniku události, jež se liší od očekávaného stavu či vývoje. Riziko by však nemělo být redukováno čistě na pravděpodobnost, protože obsahuje i šíři – dopad na daný projekt. [23]

Jedná se tedy o analýzu, která vede k odhalení a případnému snižování rizik, které hrozí v průběhu projektu. Analýzou rizik se otevírají možnosti, jak na riziko reagovat, zda ho budeme snižovat, či ho podstoupíme. Podceněním rizik může společnost přijít o finanční prostředky či ve specifických případech může dojít až ke vzniku škody na zdraví či majetku. [23]

I tato analýza je součástí diplomové práce pro snížení rizik průběhu projektu.

1.8 VOLBY A HLASOVÁNÍ

V následující kapitole se zabývám definicí konceptů volby a hlasování jakožto klíčových komponentů této diplomové práce.

1.8.1 VOLBY

Volby jsou běžný demokratický způsob, jak se dohodnout na určitých otázkách, které je potřeba řešit. Z důvodu rozličných volebních systémů a druhů voleb je těžké definovat funkci voleb. Liberálně-demokratické volby jsou převážně specifické všeobecným volebním právem a tajným hlasováním, které podporují volební soutěž.

Soutěživost ve volbách je předpokladem demokratického výběru zástupců. Harrop a Miller vysvětlují, že existují dva protichůdné názory na funkci voleb:

1. Volby jsou mechanismus, jež se zavádí do společnosti. Vyvolení lidé disponují svěřenou mocí a jsou motivování k zastupování názorů většiny (tj. jejich voličů). Zde se uplatňuje funkce voleb „zdola nahoru“ – suverenita lidu.
2. Volby jsou prostředek, pomocí kterého mohou volení vykonávat kontrolu, a tak se vytváří hierarchické uspořádání moci. Zde se uplatňuje funkce voleb „shora dolů“ – budování legitimity.

Skutečnost je taková, že volby nemají pouze jeden charakter. Volby jsou obousměrným procesem, který umožňuje voleným i voličům vzájemně se ovlivňovat. [24]

1.8.2 HLASOVÁNÍ

Hlasováním se označuje jednání jednotlivých voličů, které je ovlivněno velkým počtem faktorů. Tyto faktory, které ovlivňují volební chování, mohou být jak krátkodobé, tak dlouhodobé. Krátkodobým faktorem je například oblíbenost voleného zastupitele nebo stav momentální ekonomiky, který často reflektuje vztah mezi popularitou volených zastupitelů a ekonomickými proměnnými. Dlouhodobé faktory mohou být sociodemografické predispozice k určitému volebnímu chování, jako například pohlaví, etnicita, náboženství nebo ekonomická třída. [24]

Má volební aplikace je navržená pro hlasování a volby různých voličů, ať už se jedná o orgány vysokých škol, či jejich fakult, studenty či jiné administrátorem definované voliče. Voliči mají možnost tajně i veřejně hlasovat o různých záležitostech vysoké školy. Tyto volby mají definovaná kvora.

1.8.3 KVORUM

Každé běžné jednotlivé shromáždění by podle Henry M. Roberta (parlamentní autorita USA z 19. století) mělo mít kvorum (z latinského „quorum“, „z nichž“), které určuje kvantitativní podmínku účasti volitelů pro platnost voleb nebo hlasování. Toto číslo (nejčastěji udáváno v procentech účasti) je podmínkou pro shromáždění volitelů, aby bylo usnášeníschopné. [25]

V této práci používám pojmy „kvorum platnosti“ a „kvorum přijetí“. Kvorum platnosti je podíl voličů jako podmínka k tomu, aby bylo možno ve volbách dosáhnout výsledku, zatímco kvorum přijetí udává počet voličů vyžadovaných k tomu, aby byl výsledek voleb přijat. Kvorum přijetí může být odvozeno z celkového počtu voličů, nebo jen z hlasujících.

2 ANALÝZA PROBLÉMU A SOUČASNÉ SITUACE

V následující kapitole provedu analýzu firmy, jejích procesů a požadavků ke zpracování softwarové aplikace.

2.1 PŘEDSTAVENÍ SPOLEČNOSTI

Firma IS4U se zabývá vývojem informačního systému a svůj jediný produkt (Univerzitní informační systém = zkráceně UIS) vyvíjí již 20 let. UIS vznikl jako projekt závěrečné práce zakládajících členů společnosti na Mendelově univerzitě v Brně. Kolem roku 2010 se vývojáři UISu od Mendelové univerzity odloučili a začali nabízet produkt i jiným vysokým školám mimo Mendelu, nově již pod logem nově vzniklé společnosti IS4U. UIS začal být (a stále je) vyvíjen v jazyce Perlu. V době vzniku systému (přelomu milénia) byl Perl populární nástroj pro vývoj webových aplikací, dnes je však již tento programovací jazyk zastaralý. IS4U momentálně provozuje Univerzitní informační systém na více než deseti vysokých školách v Česku i na Slovensku. Mezi hlavní zákazníky patří: Mendelova univerzita v Brně, Vysoká škola ekonomická v Praze, Česká zemědělská univerzita v Praze, Slovenská Technická univerzita v Bratislavě a další... [26]

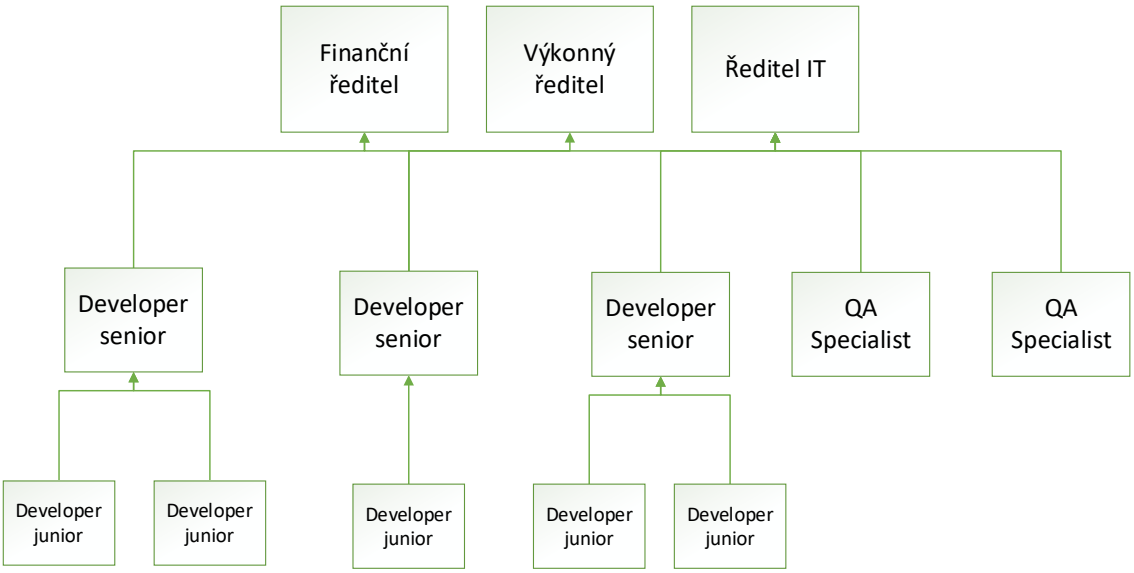
Tabulka 2: Základní informace o společnosti
(Zdroj: [27])

Název společnosti	**IS4U**
Forma obchodní společnosti	Společnost s ručením omezeným
IČO	29205336
Základní kapitál	200 000 Kč
Adresa	Roubalova 13, 602 00 Brno
Předmět podnikání	Výroba, obchod a služby neuvedené v přílohách 1 až 3 živnostenského zákona
Aktuální společníci	RNDr. Ing. Milan Šorm Ph.D., Ing. Tomáš Majer, Ing. Aleš Kutín, Mgr. Petr Dadák,

2.2 ORGANIZAČNÍ STRUKTURA

Organizační struktura firmy je lineární. Firma má tři ředitele a pět developer seniorů, pod které spadá deset developer juniorů. Jeden z developer seniorů také zastává pozici databázového architekta. Poté je ve firmě oddělení Quality Assurance čítající tři testery

pro kontrolu funkcionality vyvíjeného nebo opravovaného software a dvě sekretářky. Všech zaměstnanců je dohromady 23.



Obrázek 4: Organizační struktura firmy

(Zdroj: vlastní zpracování)

Každý developer zodpovídá za vybrané agendy. Za jednodušší agendy může zodpovídat i samotný junior (např. správa emailového subsystému, zahraniční výjezdy vyučujících), složité, kritické či obsáhlé má však na starost senior (například celé agenda studijní subsystém, správa a vyplácení stipendií, správa a generování rozvrhů). Mnohdy má jako pomocníka juniora, kde se senior stává supervizorem juniora pro danou agendu a senior řeší kritické problémy, méně kritické deleguje. Organizační struktura je v praxi víc plochá, než je formálně ve firmě zaneseno.

2.3 PRODUKTY A SLUŽBY

Jak již bylo zmíněno, společnost vyvíjí produkt Univerzitní informační systém (UIS). Jedná se o velmi rozsáhlý systém přístupný z webového prohlížeče.

Společnost čerpá finance z následujících zdrojů:

a) **Licence** – každý zákazník platí společnosti IS4U roční poplatek za provoz systému a využívání základní části UISu na své škole.

b) **Poplatky za rozšíření** – všechna rozšíření, která kdy byla naprogramována, je možné jednotlivými zákazníky dokoupit. U větších rozšíření se platí roční poplatek, u drobných rozšíření je stanovená jednorázová cena. Může se jednat o rozsáhlou agendu, či například doplnění funkcionality exportu dat pro určité tabulky. Cena se odvíjí od velikosti rozšíření.

c) **Podpora** – každá škola platí společnosti IS4U roční poplatek za podporu – tzn. společnost ručí za bezproblémový provoz systému, poskytuje poradenství a podrobnou dokumentaci pro svůj produkt. Jakmile nastane jakýkoliv incident (např. výpadek serveru školního systému), IS4U tento incident neprodleně řeší.

d) tzv. **vícepráce** – IS4U má smluvně sjednanou částku za jednu člověkohodinu práce. Pokud zákazník požádá o funkcionalitu, ovšem taková funkcionalita ještě nebyla vyvinuta, lze na zakázku tyto funkcionality doprogramovat a spustit na konkrétní instalaci.

2.4 UNIVERZITNÍ INFORMAČNÍ SYSTÉM

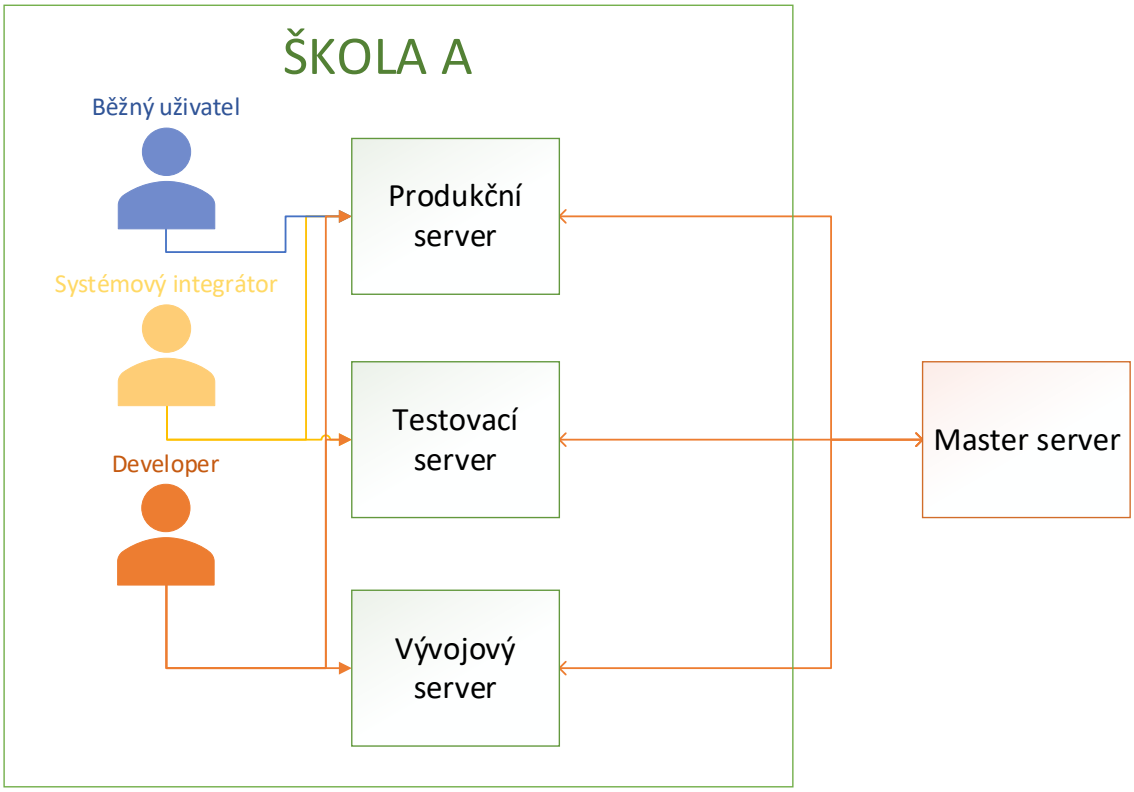
Univerzitní informační systém vznikl, jak již bylo řečeno, na půdě Mendelovy univerzity v Brně. První roky obsahoval jen základní agendy jako je studijní agenda (elektronický zápis, předměty, zkoušky, státnice). V průběhu dvaceti let se z něj stal rozsáhlý komplexní systém obsahující mimo jiné například studijní výjezdy (včetně napojení na evropskou síť Erasmus without paper), Evaluace, Business Inteligence, aplikaci Moje studium dostupné pro studenty s chytrými zařízeními Android a iOS, HR subsystém či hry pro volný čas. [28]

V současné době je systém plně přeložen do tří jazyků a na čtrnácti vysokých školách jej celkově využívá přes 130 000 uživatelů. Pro každou školu jsou v provozu tři webové domény. Pro každý skript je zaevidováno, kdo je za něj zodpovědný. Proto je možné specifikovat zodpovědnost za nalezené nedostatky ve formě reklamací či bugů. [26]

Kód Univerzitního informačního systému je na všech instalacích zákazníka udržován identický, stejně tak databázové schéma. Kód je diverzifikován Master serverem, který běží na serveru společnosti.

Každá databázová operace, která se provede v Univerzitním informačním systému, je nezaměnitelným způsobem logována. Taktéž jsou logovány i všechny http dotazy, včetně údajů o uživateli, jeho IP adresy a dalších údajů.

V následujících podkapitolách představím specifika serverů, na kterých UIS běží. Pro větší přehlednost je zobrazeno schéma přístupů jednotlivých rolí uživatelů na servery a následně vysvětleny některé podrobnosti.



Obrázek 5: Schéma přístupů jednotlivých rolí uživatelů

(Zdroj: vlastní zpracování)

Produkční server

Jedná se o hlavní server s produkční databází a produkčními skripty. Na server mají přístup všichni uživatelé. Zjednodušeně řečeno, jedná se o prostor, kde se pracuje s aktuálními a pravými daty.

Testovací server

Testovací server vznikl v reakci na možnost „otestovat si“ některé funkcionality na škole, aniž by to ovlivnilo produkční data. Skripty jsou identické na produkčním i testovacím serveru. Avšak data v databázi jsou kopií reálných dat a tato kopie se provádí jednou

týdně. Proto je možné, aby si systémový integrátor otestoval některé nové funkcionality (poté co požádá o jejich zapnutí), nebo je přímo demonstruje vedení vysoké školy, aniž by ovlivnil data na produkčním serveru.

Vývojový server

Vývojový server neběží na školních serverech, ale na serveru společnosti. Databáze vznikla kopií produkční databáze. Rozdíl oproti testovací databázi spočívá v tom, že je možné spouštět skripty upravované jednotlivými developery. Pro každou školu existuje vývojová databáze s kopií produkčních dat.

Master server

Jedná se o instalaci, která běží na serveru společnosti. Z této instalace se centrálně nastavují a upravují jednotlivá specifika pro produkční, testovací i vývojový server. Je to zejména zapínání jednotlivých modulů, agend a funkcionalit na jednotlivých školních serverech. Proto je možné, aby na všech instalacích všech škol byl stejný kód a stejné databázové schéma. Diverzifikovaný je centrálně z Master serveru.

Běžný uživatel

Pod pojmem běžný uživatel rozumíme uživatele, kteří využívají systém, ale nespravují ho. Pomocí tzv. práv, které má každý uživatel přiřazené, je systém spolu s jeho funkcemi omezen. Každá aplikace se dotazuje databáze na práva uživatele, proto například studenti nemají přístup do aplikací pro učitele. Někteří uživatelé ale mohou mít právo z více rolí. Může to být například doktorand, který studuje, ale zároveň může být cvičící, vypisovat a hodnotit zkoušky apod.

Systémový integrátor

Jedná se o zaměstnance školy, který na částečný, či plný úvazek spravuje systém. Zaměstnanci společnosti nejčastěji komunikují právě se systémovými integrátory. Tito uživatelé mají téměř plný přístup do systému, sdělují společnosti požadavky na funkcionality, reklamují chyby apod. Mají možnost vydávat se pod falešnou identitou za jiné uživatele. Toto využití je ale nezměnitelným způsobem zaznamenáno, takže při zneužití je možné tuto záměnu snadno prokázat.

Mimo produkční server mají systémoví integrátoři přístup na testovací server.

Developer

Developeři mají povětšinou omezený přístup na produkční a testovací server. Neomezený přístup mají na vývojový server. Na něm mohou využívat falešnou identitu, neomezeně si přidělovat práva atd.

2.5 INFORMAČNÍ SYSTÉM SPOLEČNOSTI

Systém společnosti sestává z jednotlivých subsystémů. V následujících podkapitolách ty nejzákladnější představím.

Helpdesk

Každý zaměstnanec má v systému Master časový harmonogram úkolů, úkoly se přidávají k určitým požadavkům a jejich náročnost je vykazována v hodinách. Systém počítá s tím, že každý zaměstnanec by měl přibližně 80 % své denní pracovní doby využít pro práci na úkolech a 20 % zbývá na jiné činnosti, jako je četba kontrolních hlášení systému o chybách a výsledcích úloh nebo e-mailovou komunikaci nevztahující se k žádnému konkrétnímu požadavku.

E-mail

Všechny komentáře, co se objeví v Helpdeskových kanálech, přijdou i jako e-mailové upozornění. E-maily se používají také pro formální komunikaci uvnitř firmy. Jako poštovního klienta společnost využívá Microsoft Outlook. Pomocí klienta se plánují schůze v zasedací místnosti nebo je využíván pro zadávání požadavků na testování v systému Redmine (o tomto systému více v následující kapitole). V kalendářích jsou zanesené dovolené, a pokud se vyskytne nějaký problém, který je nutné rychle řešit (výpadek serveru apod.), tak je také řešen pomocí e-mailové komunikace.

Redmine

V momentě, kdy má developer nachystaný kód ke zveřejnění, využije open-source systém Redmine k zadání požadavku na testování. Zákazník do tohoto systému nemá přístup a hlavní správci tohoto systému jsou z oddělení Quality Assurance. Jednotlivé požadavky mohou nabývat čtyř fází:

K otestování – jedná se o počáteční stav. Požadavek přiřazen testerům.

Vráceno k dořešení – pokud testeři naleznou nedostatky, developer musí nedostatky upravit. Požadavek přiřazen developerovi.

Otestováno – skript je připraven na vydání na všechny instalace. Požadavek je přiřazen IT řediteli.

Zveřejněno – konečný stav Redmine požadavku. IT ředitel změny zveřejnil.

Bug systém

UIS je navržen tak, aby v případě, že software vyvolá chybu, nebo výjimku[3], přišel všem developerům email s popisem chyby či doprovodnou hláškou. Pro správu těchto e-mailů společnost používá program Mutt.

Microsoft Teams

Platforma Microsoft Teams je využívána pro interní instantní komunikaci napříč všemi zaměstnanci společnosti. Maximální využití získala tato platforma zejména v době tzv. tvrdého lockdownu v průběhu celosvětové krize pandemie covid-19. Přes MS Teams byly v době vládou nařízené práce z domu realizované videokonference, které by se za normálních okolností konaly tváří v tvář.

Wiki IS4U

Wiki IS4U je web s interní dokumentací a zákazník do ní nemá přístup. Každý zaměstnanec v ní vede dokumentaci o jednotlivých agendách a zdrojových kódech.

Spolu s komentáři v kódu a Helpdesku utváří celkovou dokumentaci produktu UIS.

2.6 STRUČNÝ PŘEHLED PODNIKOVÝCH PROCESŮ

V následujících podkapitolách shrnu nejdůležitější podnikové procesy. Jsou rozdělené do dvou kategorií: hlavní a podpůrné.

2.6.1 HLAVNÍ PROCESY

Mezi hlavní procesy se řadí takové, se kterými se na denní bázi setkává většina zaměstnanců.

[3] Výjimka (anglicky „exception“) je v programování nějaká výjimečná nežádoucí situace, která nastala za běhu programu.

Návrh aplikace a tvorba zadání

Tento proces je vykonáván jak senior developery, tak junior developery. U začátku procesu stojí zákazník s požadavkem na vývoj nebo údržbu aplikace. Sdělí funkcionality a služby, které by měla aplikace nabízet. Poté developer sepíše návrh a zadání aplikace. Na návrhu a zadání se vždy podílí výkonný ředitel a databázový architekt. Poté je odhadnuta náročnost projektu (v člověkohodinách) a návrh je předložen zákazníkovi. Pokud s návrhem zákazník souhlasí, je zadání zařazeno do plánu úkolů a následuje vývoj software.

Vývoj software

Vývoj software je spolu s opravou software nejhojnějším procesem developera. Podle zadání vytvořeného během procesu „Návrh aplikace a tvorba zadání“ je aplikace vytvořena, či upravena. Samotný proces obsahuje úpravu skriptů či knihoven, objektů a jejich metod. Taktéž může obsahovat úpravu databázových tabulek, vazeb či dat.

Údržba software

Údržba software je obdobný proces, jako Vývoj software. Na začátku procesu stojí informace o nalezené chybě. Ta může pocházet od QA, zákazníka nebo mohla být vygenerovaná pomocí tzv. výjimky a obdržena pomocí e-mailu v Bug systému.

Testování software

Testování software přichází na řadu po vývoji nebo údržbě software. Oddělení QA provede testy a požadavek buď vrátí k přepracování, nebo odešle dál k synchronizaci kódu a datového schématu na všechny instalace.

2.6.2 PODPŮRNÉ PROCESY

Podpůrné procesy jsou takové procesy, které pomáhají k hladkému průběhu hlavních procesů. Mezi podpůrné procesy patří:

Úprava databázového schématu

Zejména u procesu Vývoj software dochází k nutným změnám v databázi. Nejčastěji se jedná o úpravu sloupců tabulek a vytváření nových tabulek a vazeb na stávající schéma. Tyto změny provádí databázový architekt.

Kontrola Helpdesku

Každý měsíc provádí finanční ředitel společnosti kontrolu Helpdesku. V tomto procesu jsou zkontrolovány všechny kanály, zejména termíny jednotlivých požadavků, dostatečná komunikace se zákazníkem a další. Díky kontrole Helpdesku lze zajistit, aby nedocházelo k „zapomenutí“ požadavku. Proces podporuje tok informací jak od podřízených k nadřízeným, tak od společnosti k zákazníkům.

2.7 ELEKTRONICKÉ HLASOVÁNÍ

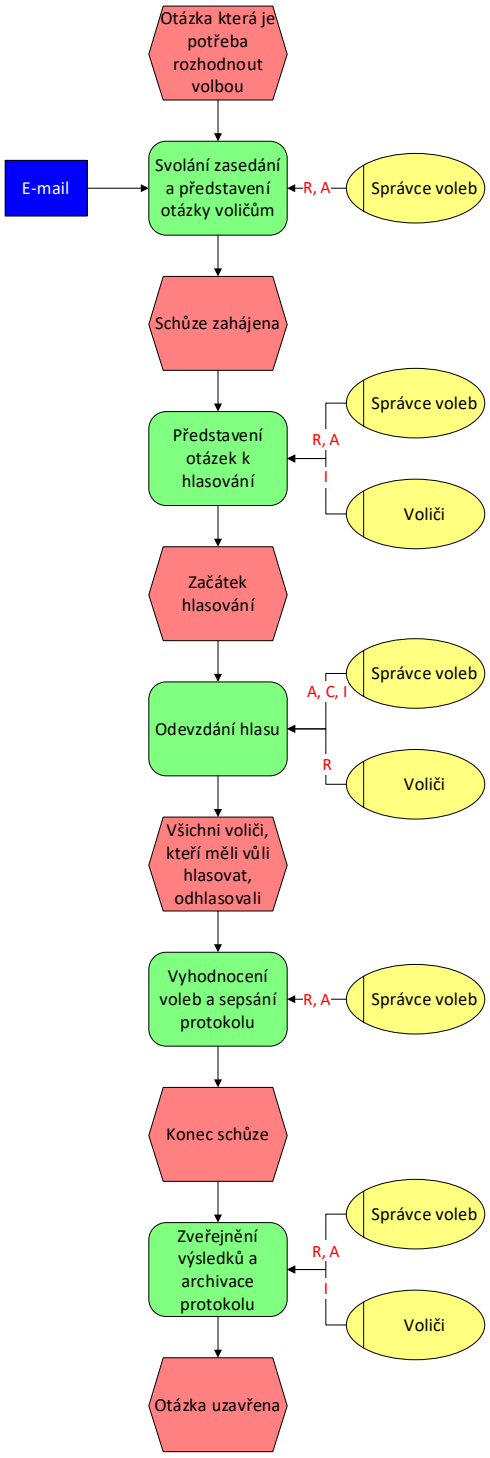
V rámci organizační struktury vysokých škol v Česku a na Slovensku působí tzv. orgány. Na jednotlivých zasedáních hlasují o mnohých záležitostech, ať už o jmenování svého předsedy, přes schvalování směrnic či rozpočtů. Mezi orgány spadá například Akademický senát, Správní rada, Vědecká rada, Disciplinární komise atd. [28] [29]

Na jaře roku 2020 začalo vedení společnosti s více zákazníky diskutovat o tom, že by bylo vhodné přenést hlasování orgánů do online prostředí. Vzešel z toho návrh na aplikaci, která by sloužila k hlasování definované množiny voličů ve volbách.

Vedení mě pověřilo k celkové záštitě daného projektu, tedy analýze volebního procesu a jeho průběhu na vysokých školách, k průzkumu trhu a návrhu řešení pro danou problematiku.

2.8 EPC DIAGRAM PROCESU HLASOVÁNÍ BEZ POUŽITÍ UIS

Hlasování na zkoumaných vysokých školách probíhá velmi podobně. Na všech zkoumaných školách probíhalo výhradně pod podmínkou fyzické přítomnosti voličů. Níže je uveden EPC diagram popisující průběh procesu hlasování. Proces hlasování nepočítá s žádnou formou korespondenčního či online hlasování, což je nejdůležitějším požadavkem na aplikaci ze strany zákazníků. K EPC diagramu je přiložena RACI matice.



Obrázek 6: EPC diagram voleb bez použití UIS

(Zdroj: vlastní zpracování)

Tabulka 3: RACI matice voleb bez použití UIS

(Zdroj: vlastní zpracování)

Činnost	Správce voleb	Volič
Svolání zasedání a představení otázky voličům	R, A	
Představení otázek k hlasování	R, A	I
Odevzdání hlasu	A, C, I	R
Vyhodnocení voleb a sepsání protokolu	R, A	
Zveřejnění výsledků a archivace protokolu	R, A	I

2.9 POŽADAVKY NA APLIKACI

Z provedené analýzy a získaných informací od zákazníka bylo zjištěno, že stávající stav hlasování není vyhovující. V Česku a na Slovensku byly školy během roku 2020 uzavřeny z důvodu pandemie koronaviru COVID-19, nejen z toho důvodu byl požadavek na elektronické hlasování aktuálnější. [30]

Cílem pro zákazníky je bezpečná, nepopiratelná online alternativa k fyzickému hlasování.

Dialogem se zákazníky a analýzou procesu hlasování vyšly najevo další skutečnosti a nároky na aplikaci, budou zde představeny:

1. Aplikace by měla umožnit elektronické hlasování jak v akademických orgánech, tak i ve vyhlašovaných referendech. Aplikace by měla umožnit jednotlivým uživatelům systému hlasovat z titulu členství v akademických orgánech, či tzv. Částech akademických obcí.[4]
2. Výstupem aplikace musí být protokol o hlasování ve formátu PDF. Na něm budou uvedeny všechny možnosti či návrhy, pro které bylo hlasováno. Volitelně půjde k protokolu přidružit seznam voličů s informací, zda hlasovali, či nikoliv.
3. Správci mohou mít tři různé role:

[4] Část akademické obce se v UISu nazývá neformálně ukotvený seznam uživatelů, který je definovaný nějakou spojující podmínkou. Příkladem může být Část akademické obce: „Všichni aktivní studenti fakulty podnikatelské“, tuto samotnou konkrétní Část akademické obce tvoří všichni aktivní studenti fakulty podnikatelské. Tyto struktury může správce informačního systému vytvářet, či upravovat.

   a. Administrátor je role správce, která zakládá, či upravuje volby, posléze volby po zveřejnění výsledků a tisku protokolu uzavírá.
   b. Volby bude možné nastavit buď tak, aby se automaticky spustily v nastavený čas, nebo pokud jsou nastaveny na manuální přepínání stavů, aby mohl operátor hlasování spustit a ukončit.
   c. Skrutátor zahajuje dešifrování hlasů, sčítání hlasů, zveřejnění výsledků voličům a tisk protokolu.
4. Je nutné, aby bylo možné vyhlásit jak návrhové kolo, tak volební kolo. Návrhové kolo je systém hlasování, kdy každý volič může navrhnout N jakýchkoliv uživatelů/slovních možností. Volební kolo je hlasování ze seznamu jednotlivých uživatelů systému či slovních možností.
5. Aplikace by měla zajistit určitou formu transparentnosti, mělo by být dohledatelné, kdo se účastnil, od kdy do kdy probíhalo hlasování, s jakým výsledkem.
6. Aplikace by měla zajistit také určitou formu tzv. „důvěryhodnosti“ – voliči mohou hlasovat jen jednou, hlas musí být prokazatelný, ale současně nesmí být dohledatelné, kdo hlasoval za předpokladu, že volba je označena jako tajná. Ani na úrovni databáze nesmí být z databáze čitelné, jak probíhá volba.
7. Ve volbách nesmí být možné vydávat se za někoho jiného (tzv. falešná identita) a nesmí být umožněno provést podvrh na úrovni databáze.
8. Operace o hlasování nesmí být zaznamenána do logu operací, tabulky přímo spojené se samotným hlasováním nesmí mít sekvenci, ID u této tabulky musí být generováno pseudonáhodně. Po ukončení hlasování se zamíchá pořadí záznamů hlasování.
9. Jednotlivá hlasování pro orgány budou moci zakládat, spouštět a vyhodnocovat předsedové a tajemníci orgánů. Pro vysokou školu bude stanoven správce, který poskytne podporu jednotlivým „zodpovědným“ osobám za chod voleb.

2.10 ANALÝZA TRHU

Po provedené analýze procesu hlasování a definici povinných funkcionalit bylo nutné zanalyzovat trh, jelikož na něm může existovat již dostupná aplikace, která by se místo vývoje aplikace nové dala implementovat.

Volebních aplikací je veliké množství, vedení společnosti a zákazníkům jsem tedy celkově představil tři předběžné návrhy. První dva návrhy jsou již existující aplikace, jejichž výhody a nevýhody shrnuji v tabulce pod danou podkapitolou. Třetí návrh je návrh na vytvoření nové aplikace pod hlavičkou IS4U.

2.10.1 HELIOSVOTING

HeliosVoting je open-source webová volební aplikace. Front end je napsán pomocí HTML a Javascriptu, backend běží v Pythonu. Pro šifrování využívá Homomorfní metody, což je poměrně nová metoda k ukládání šifrovaných dat s tou výhodou, že se dá s šifrovanými daty pracovat [31] [32].

Tabulka 4: Výhody a nevýhody aplikace HeliosVoting
(Zdroj: [31], vlastní zpracování)

HeliosVoting	
Výhody	**Nevýhody**
Aplikace je freeware	Náročná implementace
Poměrně silné zabezpečení	Nepodporuje import voličských seznamů
Open Source	Není propojen s UIS
Jednoduchost	Nejistá podpora a vývoj
Může pracovat lokálně na školních serverech	V historii měla aplikace několik bezpečnostních incidentů

2.10.2 EBALLOT

Jako další aplikaci na porovnání a jako návrh zákazníkovi jsem představil aplikaci eBallot. Jedná se o komerční aplikaci, která poskytuje opravdu rozsáhlou nabídku funkcionalit.

U voleb v systému eBallot je možné nastavit různé váhy hlasů (pokud to daná organizace umožňuje a vyžaduje), volení v zastoupení, zabezpečení je podle jejich informací na vysoké úrovni. Aplikace je poskytována jako webová služba.

Cena začíná od 2 500 USD za rok [33] (k 15.4. 2021 je to přibližně 54 250 CZK).

Tabulka 5: Výhody a nevýhody aplikace eBallot
(Zdroj: [33], vlastní zpracování)

eBallot	
Výhody	**Nevýhody**
Vysoký počet funkcionalit	Cena začíná na 54 250 Kč za rok na instituci (neboli na zákazníka)
Silné zabezpečení	Import voličských seznamů vyžaduje konverzi dat
Dobré reference	Není propojen s UIS
	Není Open source – není možné provést interní audit
	Data nebudou uložena lokálně
	Složitost na ovládání

2.10.3 NÁVRH APLIKACE SPOLEČNOSTÍ IS4U

Obě předchozí aplikace zákazníkovi silně nedostačovaly ve své nabídce. Aplikace HeliosVoting ani eBallot silně nedostačovaly specifikacím definovanými zákazníkem, proto se vedení společnosti spolu se zákazníky domluvilo na implementaci nové, vlastní aplikace, která bude navržena a implementována v rámci stávajícího Univerzitního informačního systému společností IS4U.

Jako hlavní důvod pro zavrhnutí aplikace HeliosVoting zákazník uvedl nejasnost vývoje a podpory a nutnosti vytvoření tzv. konektoru mezi systémem UIS a HeliosVoting. Systém eBallot byl zavrhnut z toho důvodu, že nebylo možné vykonat audit systému najatou společností. Druhým důvodem zamítnutí byl fakt, že data o hlasování by nebyla skladována u zákazníka, ale u americké společnosti sídlící ve Spojených státech Amerických.

Z toho důvodu jsem spolu s vedením společnosti předložil třetí variantu, tedy vlastní návrh na zpracování a implementaci všech výše uvedených bodů v kapitole požadavky na aplikaci.

V tabulce níže shrnuji výhody a nevýhody vlastního návrhu:

Tabulka 6: Výhody a nevýhody návrhu aplikace společností IS4U
(Zdroj: vlastní zpracování)

Návrh aplikace společností IS4U	
Výhody	**Nevýhody**
Funkcionality a design na míru	Vývoj potrvá déle, než při implementaci existujícího řešení
Plné propojení se systémem UIS	Provoz mohou provázet problémy, nový systém nemusí být plně odladěn
Data budou uložena lokálně	
Cena bude ovlivněna počtem zapojených vysokých škol	
Silné zabezpečení	

2.11 VÝSTUPY ANALÝZ

V kapitole *Analýza problému a současné situace* jsem analyzoval firmu, její systémy a požadavek zákazníka na volební aplikaci. Součástí byla analýza trhu s volebními aplikacemi a tyto aplikace jsem následně předložil vedení společnosti IS4U a zákazníkovi. Po jejich vzájemné domluvě jsem byl pověřen návrhem, tvorbu a implementací volební aplikace v rámci systému UIS.

Stávající analýzy mi budou podkladem pro následující kapitoly práce.

3 VLASTNÍ NÁVRHY ŘEŠENÍ, PŘÍNOS NÁVRHŮ ŘEŠENÍ

Tato část práce je zaměřena na shrnutí požadavků na vývoj aplikace, nastavení milníků, samotný návrh a vývoj aplikace a následné zhodnocení.

3.1 MILNÍKY VÝVOJE MODULU

Pro vývoj webové aplikace je vhodné využít určitý postup, protože některé části na sebe logicky navazují. Je například důležité psát backend aplikace až ve chvíli, kdy je připravena databáze. Vytyčím tedy základní milníky vývoje aplikace:

- Analýza rizik
- Časová analýza
- Návrh elektronického hlasování v UIS
- Tvorba aplikace
- Testování aplikace
- Zveřejnění aplikace
- Školení zákazníků
- Testovací provoz (testování zákazníkem)
- Ostrý provoz a poprojektová fáze

3.2 ANALÝZA RIZIK

V následujících podkapitolách bude provedena analýza rizik a případné snižování, či podstoupení nalezených rizik.

3.2.1 METRIKY PRO HODNOCENÍ RIZIKA

V následujících tabulkách jsou definovány škály pro hodnocení rizika. Jedná se o číselné a slovní vyjádření míry pravděpodobnosti a dopadu rizika. Z toho vyplývá následná hodnota rizika.

Tabulka 7: Míra pravděpodobnosti výskytu rizika

(Zdroj: vlastní zpracování)

Míra pravděpodobnosti	
Téměř žádná	1–2 (0 % - 19 %)
Nízká	3–4 (20 % - 39 %)
Pravděpodobná	5–6 (40 % - 59 %)
Více pravděpodobná	7–8 (60 % - 79 %)
Vysoce pravděpodobná	9–10 (80 % - 100 %)

Tabulka 8: Dopad rizika na implementaci

(Zdroj: vlastní zpracování)

Dopad	
Minimální	1–2
Nízký	3–4
Významný	5–6
Velmi významný	7–8
Kritický	9–10

Hodnota rizika je součinem míry pravděpodobnosti a dopadu. Z toho vyplývá poslední závislost:

Tabulka 9: Hodnota rizika a jeho významnost

(Zdroj: vlastní zpracování)

Hodnota rizika	Významnost
0–25	Nízká
26–50	Střední
51–75	Vysoká
76–100	Kritická

3.2.2 IDENTIFIKACE HROZEB A SCÉNÁŘŮ

Po komunikaci s vedením podniku, rešerši častých rizik v IT projektech a doplnění rizik plynoucích z vlastních zkušeností s projekty byla nalezena následující rizika a vybrána ta, která by významně mohla ovlivnit průběh implementace. Tato rizika ohodnotím pomocí skórovací metody. Navrhnu opatření a znovu zmapuji rizika po aplikaci opatření. [34]

Z následující tabulky je názorné, že rizika s nejvyšší hodnotou vyplývají z hrozby zahlcení oddělení IT mnoha procesy (R6) nebo že aplikace bude obsahovat chyby, které vyústí v nepoužitelnost daného softwaru (R13).

Tabulka 10: Kvantifikace rizika

(zdroj: vlastní zpracování)

Č. rizika	Hrozba	Scénář	Míra pravděp.	Dopad	H. rizika
R1	Špatně provedený návrh.	Bude možné zjistit, kdo a jak hlasoval.	4	10	**40**
R2	Návrh bude jednoúčelový.	Bez možnosti, či s obtížemi implementovatelný do jiných agend či ostatním zákazníkům.	4	9	**36**
R3	Návrh nebude dostatečně přesný.	Produkt bude odchýlen od záměru zákazníka, náklady na reklamaci.	4	8	**32**
R4	Projekt nebude disponovat developery s požadovanými dovednostmi.	Termín projektu se zpozdí, náklady se prodraží.	4	8	**32**
R5	Dodatečná nepřítomnost klíčových zaměstnanců v průběhu trvání projektu.	Projekt se prodraží, software bude obsahovat chyby, termín se zpozdí.	6	7	**42**
R6	Oddělení IT bude zahlceno mnoha procesy.	Termín projektu se zpozdí, aplikace bude obsahovat chyby.	8	7	**56**
R7	Špatně provedená časová analýza.	Nestihne se termín projektu.	6	5	**30**
R8	Vedení nebude počítat se "špatným scénářem" časové analýzy.	Projekt se prodraží, termín se zpozdí.	4	8	**32**
R9	Vedení nebude mít přehled o průběhu projektu.	Termín projektu se zpozdí.	5	5	**25**
R10	Nebude kladen důraz na činnosti ležící na kritické cestě.	Termín projektu se zpozdí.	7	5	**35**
R11	Nevhodně nastavená databáze.	Zákazník přijde o data hlasování.	4	9	**36**
R12	Riziko ztráty dat zdrojového kódu.	Termín projektu se zpozdí, náklady se prodraží.	4	10	**40**
R13	Nová aplikace bude obsahovat chyby, nebude možné ji využívat.	Poškození dobrého jména firmy, reklamace aplikace, termín projektu se zpozdí.	5	10	**50**
R14	Zákazník nebude spokojen s předaným produktem.	Poškození dobrého jména firmy, náklady na dodatečnou podporu.	6	8	**48**
R15	Zákazník nebude umět aplikaci používat.	Poškození dobrého jména firmy, náklady na dodatečnou podporu.	5	7	**35**

Mnohá rizika by mohla vést ke zpoždění termínu projektu či poškození dobrého jména firmy.

3.2.3 SNIŽOVÁNÍ RIZIK

Následují opatření, která snižují rizika. Podstoupení rizika nebylo vhodné navrhnout ani u jednoho rizika. U rizik s nejvyšší hodnotou navrhuji důraz na výběr vhodného termínu, během kterého nebudou probíhat větší projekty, aby byla vyšší operativní kapacita lidských zdrojů (R6). U R13 návrh počítá s Code Review od ředitele IT. Ten ověří hlavně logické chyby v klíčových procesech po vytvoření kódu aplikace. Po Code Review proběhne důkladné testování ze strany oddělení QA.

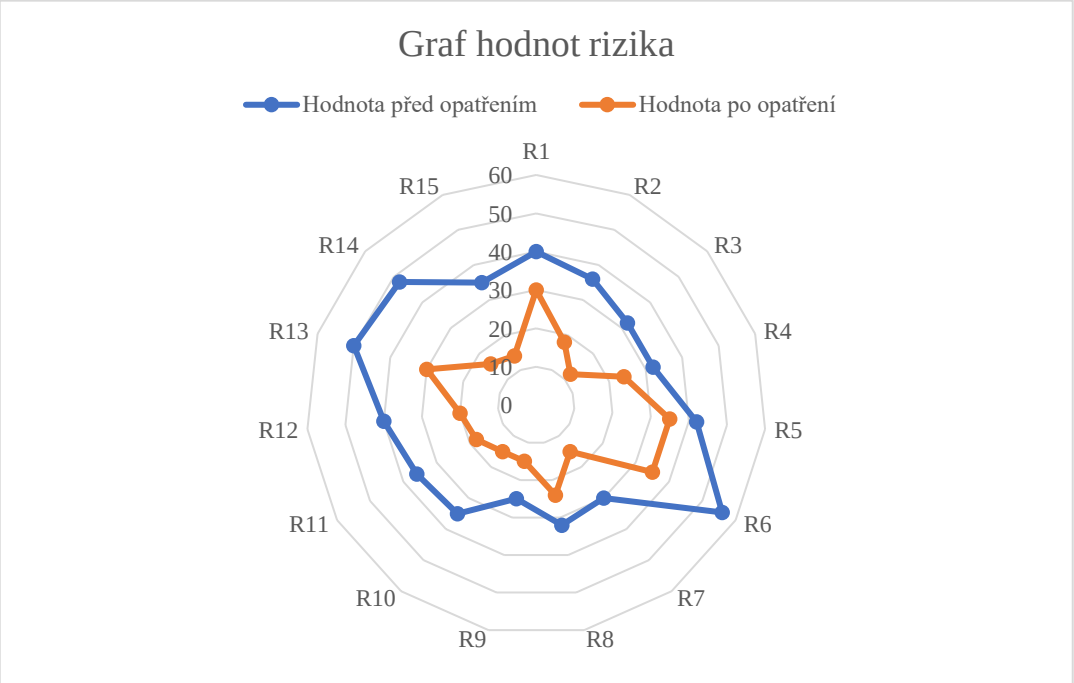
Tabulka 11: Snižování rizik

(Zdroj: vlastní zpracování)

Číslo rizika	Návrh opatření	Nová míra p.	Nový dopad	Nová hodnota rizika
R1	Při návrhu bude kladen důraz na anonymitu hlasování, Ředitel IT provede Design Review.	3	10	**30**
R2	Proběhne schůze se zodpovědnými z jiných agend, požadavek na aplikaci se bude konzultovat jedním kanálem se všemi zákazníky.	2	9	**18**
R3	Budou naplánované dvě průběžné schůze ohledně návrhu se zákazníky.	3	4	**12**
R4	Tvorba se naplánuje na vhodný termín, nebude probíhat zároveň s podobně velkým projektem.	3	8	**24**
R5	Meetingy budou naplánovány online, bude naplánována dostatečná časová rezerva.	5	7	**35**
R6	Tvorba se naplánuje na vhodný termín, nebude probíhat zároveň s podobně velkým projektem.	5	7	**35**
R7	Každý týden bude schůze s vedením ohledně odvedené práce.	3	5	**15**
R8	Bude stanovena rezerva v navržené ceně za produkt.	3	8	**24**
R9	Každý týden bude schůze s vedením ohledně odvedené práce.	3	5	**15**
R10	Každý týden bude schůze s vedením ohledně odvedené práce.	3	5	**15**
R11	Správce databáze nastaví zálohování klíčových tabulek agendy elektronického hlasování.	2	9	**18**
R12	Kód aplikace bude vyvíjen a uložen výhradně na úložišti firemního serveru, který je pravidelně zálohován.	2	10	**20**
R13	Technický ředitel provede Code Review na snížení rizika logických chyb, aplikace bude důkladně otestována oddělením QA.	3	10	**30**
R14	Budou naplánované dvě průběžné schůze ohledně návrhu se zákazníky.	2	8	**16**
R15	Proběhne školení zákazníků s důrazem na hlavní funkcionality aplikace.	2	7	**14**

3.2.4 ZHODNOCENÍ ANALÝZY RIZIK

Pomocí snižování rizik jsem docílil toho, že se hodnota všech rizik snížila, tudíž průměrná hodnota analyzovaných rizik (R1 až R15) klesla z původních přibližných 38 na přibližných 21. Průměrná hodnota rizika je před návrhem a po návrhu opatření nízká.



Obrázek 7: Graf hodnot rizika

(Zdroj: vlastní zpracování)

3.3 ČASOVÁ ANALÝZA

Pro časovou analýzu jsem využil metodu PERT. Ta je v rámci této práce vhodnější než metoda CPM, protože je vhodná pro projekty s nepřesnou délkou trvání. Pro metodu PERT jsem použil činnosti od návrhu přes tvorbu a implementaci. Činnosti uzavírá poprojektová fáze. Pro určení délky trvání jsem použil expertní odhad, zkušenost z předchozích podobných projektů a komunikaci s kolegy. [28]

Následuje časová analýza PERT – tabulka a graf. V grafu je červeně vyznačena kritická cesta. Čas je v analýze vykazován ve dnech, respektive člověkodnech.

Tabulka 12: Časová analýza PERT

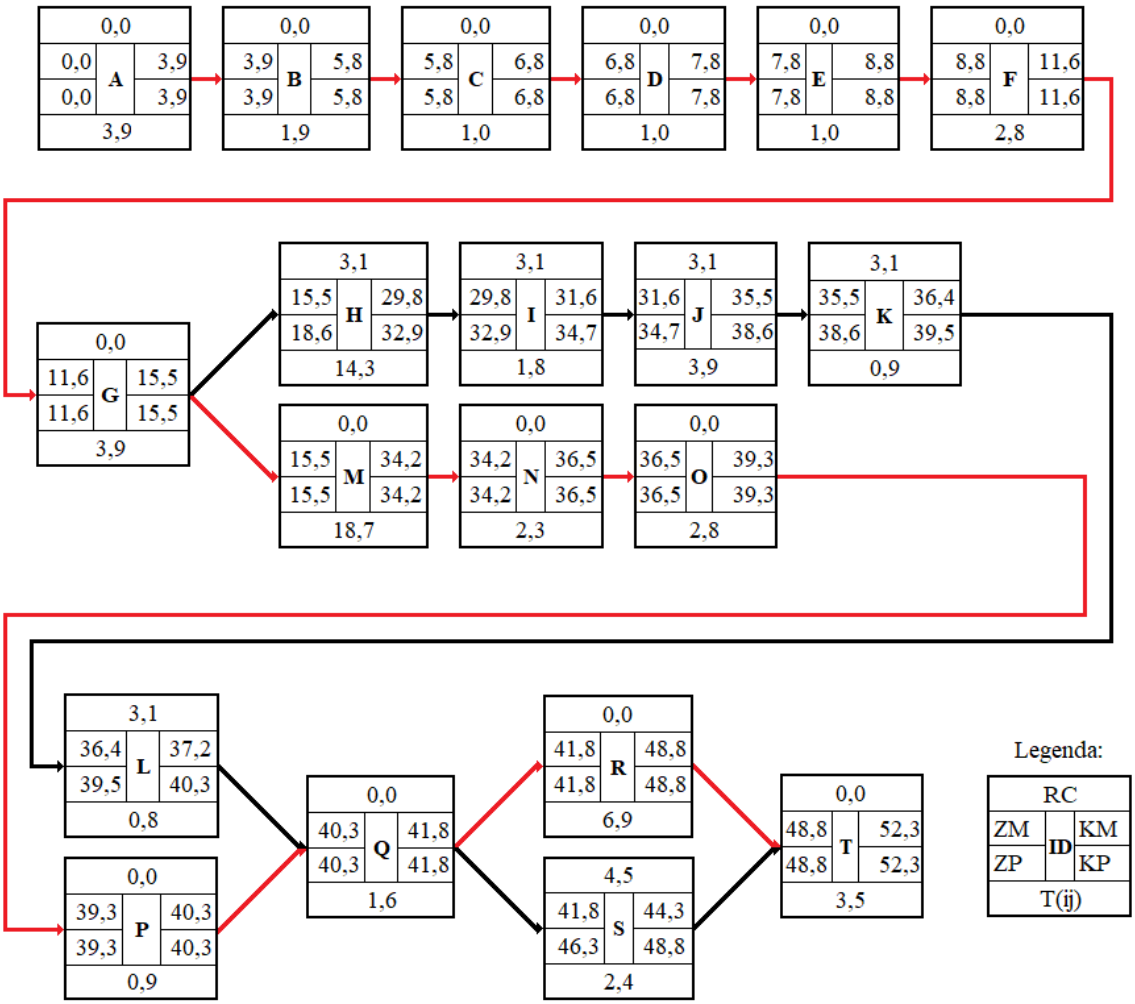
(Zdroj: vlastní zpracování)

ID	Popis činnosti	i	j	a[ij]	m[ij]	b[ij]	t[ij]	$\sigma[ij]$	$\sigma 2[ij]$	ZM	KM	ZP	KP	RC
A	Návrh elektronického hlasování v UIS	_	B	2,5	4	5	3,9	1	1,6	0	3,9	0	3,9	**0**
B	Design Review	A	C	1	2	2,5	1,9	0,6	0,6	3,9	5,8	3,9	5,8	**0**
C	Výběr a volba technologie	B	D	0,5	1	1,5	1,0	0,4	0,3	5,8	6,8	5,8	6,8	**0**
D	Schůze se zákazníky k diskusi nad návrhem	C	E	0,5	1	1,5	1,0	0,4	0,3	6,8	7,8	6,8	7,8	**0**
E	Schůze s kolegy a vedením	D	F	0,5	1	1,5	1,0	0,4	0,3	7,8	8,8	7,8	8,8	**0**
F	Návrh datové struktury	E	G	1	2,5	5,5	2,8	1,9	5,3	8,8	11,6	8,8	11,6	**0**
G	Návrh backendu a frontendu	F	HM	2,5	4	5	3,9	1,0	1,6	11,6	15,5	11,6	15,5	**0**
H	Tvorba aplikací pro správce	G	I	10	14	20	14,3	4,1	25,3	15,5	29,8	18,6	32,9	**3,1**
I	Code Review aplikací pro správce	H	J	0,5	1,5	4	1,8	1,5	3,3	29,8	31,6	32,9	34,7	**3,1**
J	Testování aplikací pro správce	I	K	2	4	5,5	3,9	1,4	3,1	31,6	35,5	34,7	38,6	**3,1**
K	Schůze se zákazníky k diskusi nad průběhem	J	L	0,5	1	1	0,9	0,2	0,1	35,5	36,4	38,6	39,5	**3,1**
L	Zveřejnění aplikací pro správce	K	Q	0,5	0,75	1	0,8	0,2	0,1	36,4	37,2	39,5	40,3	**3,1**
M	Tvorba aplikací pro voliče	G	N	15	18	25	18,7	4,2	26,3	15,5	34,2	15,5	34,2	**0**
N	Code Review aplikací pro voliče	M	O	1	2	5	2,3	1,7	4,3	34,2	36,5	34,2	36,5	**0**
O	Testování aplikací pro voliče	N	P	1	2,5	6	2,8	2,1	6,6	36,5	39,3	36,5	39,3	**0**
P	Zveřejnění aplikací pro voliče na testování	O	Q	0,5	1	1	0,9	0,2	0,1	39,3	40,3	39,3	40,3	**0**
Q	Školení zákazníků	LP	RS	1	1,5	2,5	1,6	0,6	0,6	40,3	41,8	40,3	41,8	**0**
R	Testovací provoz (testování zákazníkem)	Q	T	5	6	12,5	6,9	3,3	16,6	41,8	48,8	41,8	48,8	**0**
S	Sepsání dokumentace	Q	T	1,5	2,5	3	2,4	0,6	0,6	41,8	44,3	46,3	48,8	**4,5**
T	Poprojektová fáze	RS	_	2	3,5	5	3,5	1,2	2,3	48,8	52,3	48,8	52,3	**0**

Z tabulky i grafu je patrné, že mimo kritickou cestu leží jen činnosti, které se týkají vývoje aplikací pro správce (které jsou méně časově náročné, než aplikace pro voliče) a sepsání dokumentace.

Činnosti ležící na kritické cestě jsou prioritní – zpoždění má za následek prodloužení trvání celého projektu.

Časová náročnost návrhu, tvorby a implementace aplikace je pomocí metody PERT vypočtena na 76,3 člověkodní. Délka trvání projektu bude při dostatečných kapacitách a dodržení všech termínů 52,3 dne.



Obrázek 8: PERT graficky

(Zdroj: vlastní zpracování)

Začátek projektu byl stanoven na 1. června 2021. Následuje harmonogram činností dle analýzy PERT. Z tabulky Harmonogram projektu je patrné, že projekt bude při dodržení termínů ukončen 23. července 2021.

Tabulka 13: Harmonogram projektu

(Zdroj: vlastní zpracování)

ID	Popis činnosti	ZM	KM	RC
A	Návrh elektronického hlasování v UIS	01.06.2021	04.06.2021	0,0
B	Design Review	04.06.2021	06.06.2021	0,0
C	Výběr a volba technologie	06.06.2021	07.06.2021	0,0
D	Schůze se zákazníky k diskusi nad návrhem	07.06.2021	08.06.2021	0,0
E	Schůze s kolegy a vedením	08.06.2021	09.06.2021	0,0
F	Návrh datové struktury	09.06.2021	12.06.2021	0,0
G	Návrh backendu a frontendu	12.06.2021	16.06.2021	0,0
H	Tvorba aplikací pro správce	16.06.2021	30.06.2021	3,1
I	Code Review aplikací pro správce	30.06.2021	02.07.2021	3,1
J	Testování aplikací pro správce	02.07.2021	06.07.2021	3,1
K	Schůze se zákazníky k diskusi nad průběhem	06.07.2021	07.07.2021	3,1
L	Zveřejnění aplikací pro správce	07.07.2021	08.07.2021	3,1
M	Tvorba aplikací pro voliče	16.06.2021	05.07.2021	0,0
N	Code Review aplikací pro voliče	05.07.2021	07.07.2021	0,0
O	Testování aplikací pro voliče	07.07.2021	10.07.2021	0,0
P	Zveřejnění aplikací pro voliče na testování	10.07.2021	11.07.2021	0,0
Q	Školení zákazníků	11.07.2021	12.07.2021	0,0
R	Testovací provoz (testování zákazníkem)	12.07.2021	19.07.2021	0,0
S	Sepsání dokumentace	12.07.2021	15.07.2021	4,5
T	Poprojektová fáze	19.07.2021	23.07.2021	0,0

3.4 NÁVRH ELEKTRONICKÉHO HLASOVÁNÍ V UIS

Po analýze a vyjasnění všech předběžných požadavků ze strany zákazníka, analýze rizik a analýze časové byl vytvořen návrh aplikace pro elektronické hlasování.

Cílovým stavem tohoto procesu je vyhodnocené hlasování a volební protokol, tisknutelný PDF soubor s výsledkem voleb.

Tabulka 14: Životní cyklus voleb, stavy

(Zdroj: vlastní zpracování, [35])

Poř.	Ikona	Název stavu
1.		Probíhá příprava
2.		Volby připraveny
3.		Probíhá hlasování
4.		Hlasování ukončeno
5.		Volební výsledky sečteny
6.		Volební výsledky zveřejněny
7.		Volby uzavřeny

3.4.1 ŽIVOTNÍ CYKLUS VOLEB

V následující kapitole jsou rozepsány jednotlivé stavy voleb a popsán cyklus od založení voleb, přes hlasování, až po vyhodnocení a archivaci.

Stav: Probíhá příprava

Na počátku voleb stojí otázka, kterou je potřeba hlasováním rozhodnout. Proto administrátor voleb vytvoří volby. Vloží název voleb, předmět hlasování, okruh voličů, fázi voleb (Volební kolo/Návrhové kolo), druh voleb (Výběr/Návrh z uživatelů, či Výběr/Návrh z možností), počet možností či návrhů, které mohou voliči vyplnit. Poté určí datum a čas začátku a konce hlasování, nastaví kvorum a přidá instrukce voličům.

Stránka správci voleb automaticky vygeneruje pro volby pár klíčů: veřejný a soukromý. Soukromý klíč je možné ponechat v systému, buď zašifrovaný nebo nezašifrovaný. Veřejný klíč zůstává uložen v systému tak, jak je.

Po založení voleb je možné upravit voličský seznam, nadefinovat správce s příslušnými rolemi či možnosti, ze kterých budou voliči vybírat. Poté administrátor potvrdí nastavení voleb a přepne volby do stavu „Volby připraveny“.

Stav: Volby připraveny

V nastavený čas (nebo ručně) se spustí hlasování. Voliči jsou o spuštění hlasování informováni e-mailem. Pro každého voliče systém vygeneruje jeden prázdný hlas v tabulce Pool hlasů[5]. To zvýší bezpečnost a sníží dohledatelnost toho, kdo hlasoval.

Stav: Probíhá hlasování

Během stanovené doby probíhá hlasování. Ve voličské aplikaci bude mít uživatel možnost prohlížet volby, ve kterých je voličem. Tyto volby uvidí od okamžiku spuštění hlasování. Bude mít možnost prohlížet výsledky uplynulých voleb. Pokud se rozhodne hlasovat, otevře si volební lístek, který se skládá z následujících částí:

- Základní informace o volbách (začátek hlasování, konec hlasování, informace, zda se volič může zdržet hlasování, seznam voličů)
- Předmět hlasování
- Instrukce voličům
- Volební formulář (dle nastavení voleb zde mohou být vstupní prvky HTML formuláře jako radio, checkbox nebo text)
- Tlačítko Hlasovat pro potvrzení

Po stisknutí tlačítka „Hlasovat“ skript napsaný v jazyce Javascript zašifruje Volební formulář a vytvoří unikátní volební hlas prostorovým složením šifrovaného klíče a inicializačního vektoru. Tento unikátní volební hlas má vypočten digest SHA-256 a je zobrazen uživateli k zapsání. Teprve po tomto potvrzení je šifrovaný hlas v podobě hexadecimálního řetězce zaslán na server a server hlas uloží na náhodné místo v tabulce Poolu hlasů.

[5] „Volby_pool_hlasu“ je tabulka, která představuje jakousi volební urnu. Jsou v ní uchovány volební hlasy v zašifrované podobě.

Stav: Hlasování ukončeno

Po uplynutí doby hlasování (nebo pokud odhlasuje poslední volič) systém sám ukončí hlasování. Poté je e-mailem informován skrutátor, aby provedl „zamíchání, dešifrování a sčítání voleb“.

Stav: Volební výsledky sečteny

Skrutátor zobrazí volby v aplikaci pro volební výsledky. Po odsouhlasení webový prohlížeč skrutátora v prohlížeči dešifruje všechny volební hlasy pomocí soukromého klíče. Na serveru jsou výsledky zkontrolovány na platnost parametrů ve volebním formuláři a naplněny do tabulek pro výsledky. Po zapsání výsledků do databáze je skrutátorovi zobrazena tabulka s výsledky. Je také možné vytisknout si protokol s výsledky voleb.

Stav: Volební výsledky zveřejněny

Po zveřejnění výsledků jsou výsledky dostupné i pro voliče. Ti v této aplikaci nemají možnost tisknout protokol, ale mohou se podívat, jak dané hlasování dopadlo.

Stav: Volby uzavřeny

Po uvážení administrátor voleb tyto volby tzv. „Uzavře“. Nejedná se o nic jiného, než že se primárně skryjí ze seznamů voleb. Tato funkcionalita slouží pouze k tomu, aby byla aplikace přehlednější.

3.4.2 EPC DIAGRAM

EPC Diagram vytvořený pro volení v systému UIS je z důvodu rozměrů v příloze na konci práce na stranách II až IV (protože je rozdělen na 3 strany). Diagram zachycuje průběh voleb od návrhu, přes hlasování až po uzavření voleb. EPC diagram usnadní orientaci developera při vytváření aplikace.

3.4.3 RACI MATICE

Pro doplnění EPC diagramu jsem sestavil RACI matici pro definici odpovědnosti.

Tabulka 15: RACI matice procesu volení

(Zdroj: vlastní zpracování)

Aktivita	Administrátor	Operátor	Skrutátor	Volič
Vytvoření návrhu voleb	R, A			
Definice možností	R, A			
Definice voličů	R, A			
Spuštění hlasování	C, I	R, A		I
Hlasování				R, A
Ukončení hlasování	C, I	R, A		I
Dešifrování a sčítání hlasů	C, I		R, A	
Tisk protokolu a archivace	C, I		R, A	
Zveřejnění výsledků voličům	C, I		R, A	I
Uzavření voleb	R, A			

RACI matice spolu s EPC diagramem je důležitým podkladem pro návrh a vytvoření aplikace. Z RACI matice je názorné, ve kterých situacích je volič informován o průběhu voleb. Taktéž je poskytnut stručný přehled pravomocí jednotlivých rolí správce a popis situace, kdy jednotlivé role správců jsou rozděleny mezi více uživatelů.

3.4.4 VÝBĚR A VOLBA TECHNOLOGIE

Volební proces hlasování vyžaduje zapojit všechny běžné prvky UIS. Jelikož drtivá většina aplikací je v UIS napsána v jazyce Perl, bude nejvhodnější pro kompatibilitu s celým systémem použít právě jazyk Perl. Databáze, kterou systém UIS využívá, je Oracle Database 12c Enterprise Edition Release 12.1.0.2.0. Jazyk Javascript bude využit pro šifrování hlasovacího lístku a dešifrování balíku hlasovacích lístků přímo v prohlížeči.

3.4.5 NÁVRH DATOVÉ STRUKTURY

Před každou novou aplikací je vhodné zanalyzovat, jaká data budu používat a případně jaké tabulky s jakými vazbami budu potřebovat.

Jako první tabulku jsem v návrhu uvažoval tabulku Volby. Jedná se o tabulku, ve které budou uloženy všechny parametry voleb – název, předmět hlasování, od kdy do kdy se volby budou konat, dále privátní a veřejný klíč na šifrování a informace ohledně definice

voličů (u voleb bude možné nadefinovat, že se jedná o „Akademický senát“ – voličský seznam se při tomto typu voleb naplní automaticky…).

Další stěžejní tabulka byla při návrhu „Volby_volici“. Jedná se o seznam voličů. Voliči můžou být interní (tzn. zaevidovaní uživatelé systému), nebo externí (definováni celým jménem voliče a jeho e-mailovou adresou). Pro externí voliče se při evidování generuje jednorázový přístupový hexadecimální řetězec, jehož znalost (předána v HTML parametru) jednoznačně identifikuje voliče a umožní mu hlasování (respektive přístup do hlasovacích aplikací).

Mimo to je nutné definovat „Volby_spravci“ – správce pro jednotlivé volby a jejich role – Administrátor, Operátor a Skrutátor.

„Volby_moznosti“ budou sloužit výhradně pro zápis možností, ze kterých mohou voliči volit. Tabulka je navržena tak, aby se mohlo vybírat z uživatelů systému, nebo z textu, jako je třeba „první patro“ či „F“ jako známka.

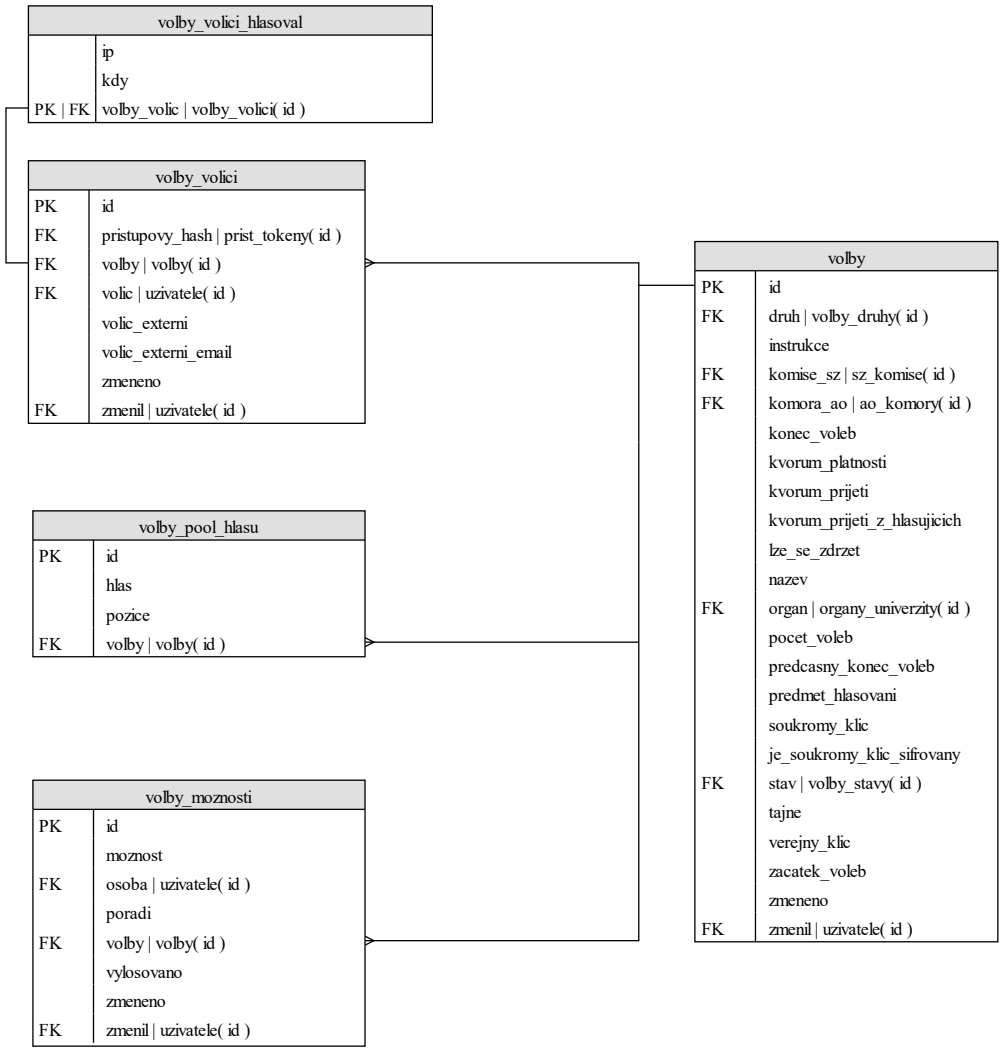
Tabulka 16: Seznam tabulek a jejich popis

(Zdroj: vlastní zpracování)

Název tabulky	Popis
Volby	Základní objekt Volby
Volby_druhy	Číselník druhu voleb
Volby_role	Číselník rolí správců voleb
Volby_faze	Číselník fáze voleb
Volby_nepl_hlasy_stavy	Číselník stavu neplatných hlasů
Volby_stavy	Číselník stavu, ve kterém se volby nacházejí
Volby_moznosti	Možnosti voleb
Volby_vysledky_moznosti	Tabulka propojující Hlas a možnosti voleb
Volby_spravci	Správci voleb
Volby_neplatne_hlasy	Neplatné hlasy vyřazené při dešifrování a sčítání
Volby_vysledky	Volební výsledky obsahuje dešifrovaný hlas s vazbou na možnosti
Volby_volici	Voličský seznam
Volby_pool_hlasu	Pool hlasů obsahující hlasy v zašifrované podobě
Volby_volici_hlasoval	Jednosloupcová tabulka s ID voliče, indikující, zda volič hlasoval
Organy_univerzity	Tabulka orgánů vysoké školy (pro propojení agend)
Sz_komise	Tabulka státnicových komisí (pro propojení agend)
Ao_komory	Tabulka částí akademické obce (pro propojení agend)
Prist_tokeny	Tabulka s přístupovými tokeny pro hlasování externích voličů

Tabulky pro hlasování

Tabulky pro hlasování popíšu podrobněji, protože jsou podstatné pro pochopení nejdůležitějších částí procesu voleb. Při přepnutí voleb ze stavu Probíhá příprava do stavu Volby připraveny se naplní tabulka „Volby_pool_hlasů“ – pro každého voliče jeden záznam s prázdným textovým polem datového typu CLOB ve sloupci hlas. Po spuštění voleb volič v aplikaci s názvem „Volební lístek“ zaškrtne možnosti, pro které se rozhodl hlasovat a stiskne tlačítko „zašifrovat volební lístek“ – Javascriptová funkce zanalyzuje formulář a vytvoří z něj hexadecimální řetězec, který v zašifrované podobě předá do HTML parametru. Poté je možné stisknout tlačítko „Odeslat hlasovací lístek“ a následně v jedné transakci proběhne naplnění náhodného prázdného záznamu v tabulce „Volby_pool_hlasu“ a „Volby_volici_hlasoval“.



Obrázek 9: Schéma tabulek pro hlasování

(Zdroj: vlastní zpracování)

Takto jsou postupně zaznamenávány hlasy v tabulce „Volby_pool_hlasu“ během celého hlasování.

Dešifrování a sčítání voleb skrutátorem

Poté, co se ukončí hlasování, skrutátor dešifruje volby a tyto hlasy se zapíšou do již klasických tabulek. Z tabulky „Volby_pool_hlasu“ se naplní tabulky „Volby_vysledky_moznosti“ a „Volby_vysledky“. Jelikož server při hlasování dostává hlasy v zašifrované podobě, není možné eliminovat neplatné hlasy – různě namnožené a upravené parametry HTML formuláře. Existuje sice Javascriptová kontrola úpravy těchto parametrů, ale JavaScript se dá poměrně snadno obejít, či úplně vypnout. Proto se serverová kontrola provádí až při sčítání hlasů – z toho důvodu existuje tabulka „Volby_neplatne_hlasy“, kde se případné dešifrované neplatné hlasy uloží.

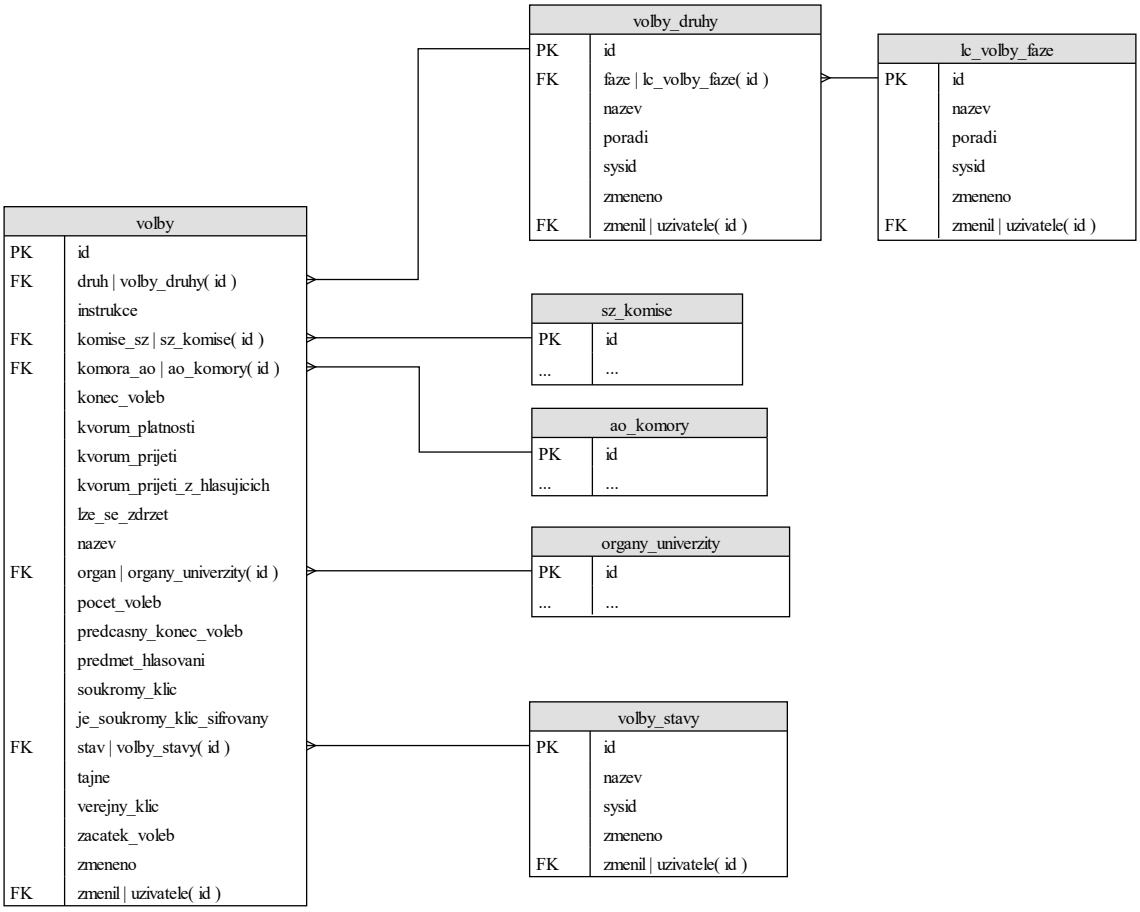
Detail tabulek, které se používají k dešifrování a sečtení hlasů vypadá následovně:

Obrázek 10: Dešifrování a sčítání voleb skrutátorem

(Zdroj: vlastní zpracování)

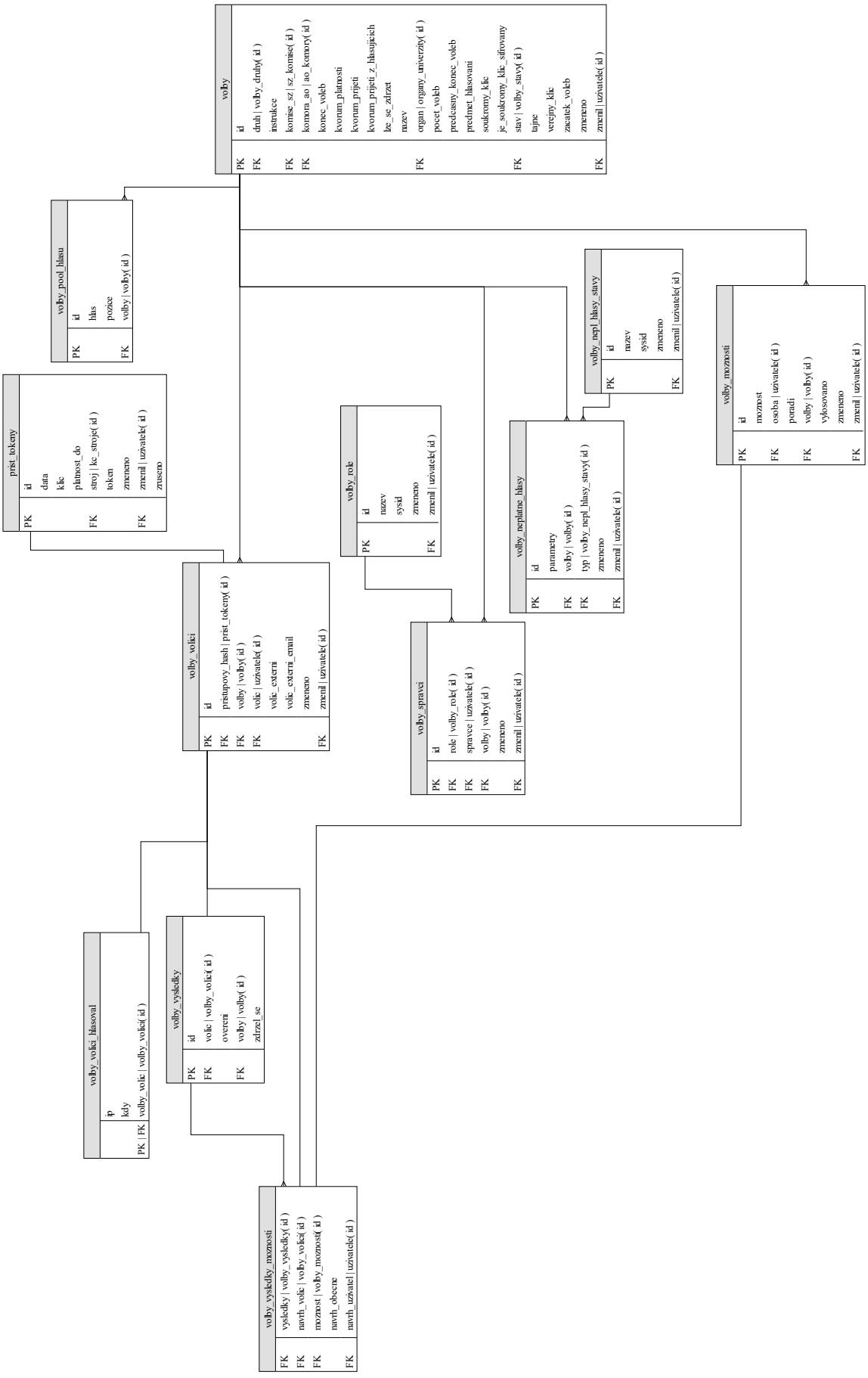
Tabulka Volby je také navázána na některé další agendy systému UIS– například agenda státnic, či orgánů atd. Volby se dají zakládat i pro státnice, kde aplikace automaticky vygeneruje pro každého studenta jedny volby – voliči jsou státnicová komise a možnosti jsou známková škála ECTS či „složil/nesložil“. Z důvodu rozsahu a tématu práce se napojením do jiných agend budu zabývat jen okrajově.

Následuje celé schéma voleb. Z důvodu velikosti jej bylo nutné rozdělit na dvě části. V celku je na konci této práce v příloze.



Obrázek 11: ERD Diagram 1. část

(Zdroj: vlastní zpracování)



Obrázek 12: ERD Diagram 2. část

(Zdroj: vlastní zpracování)

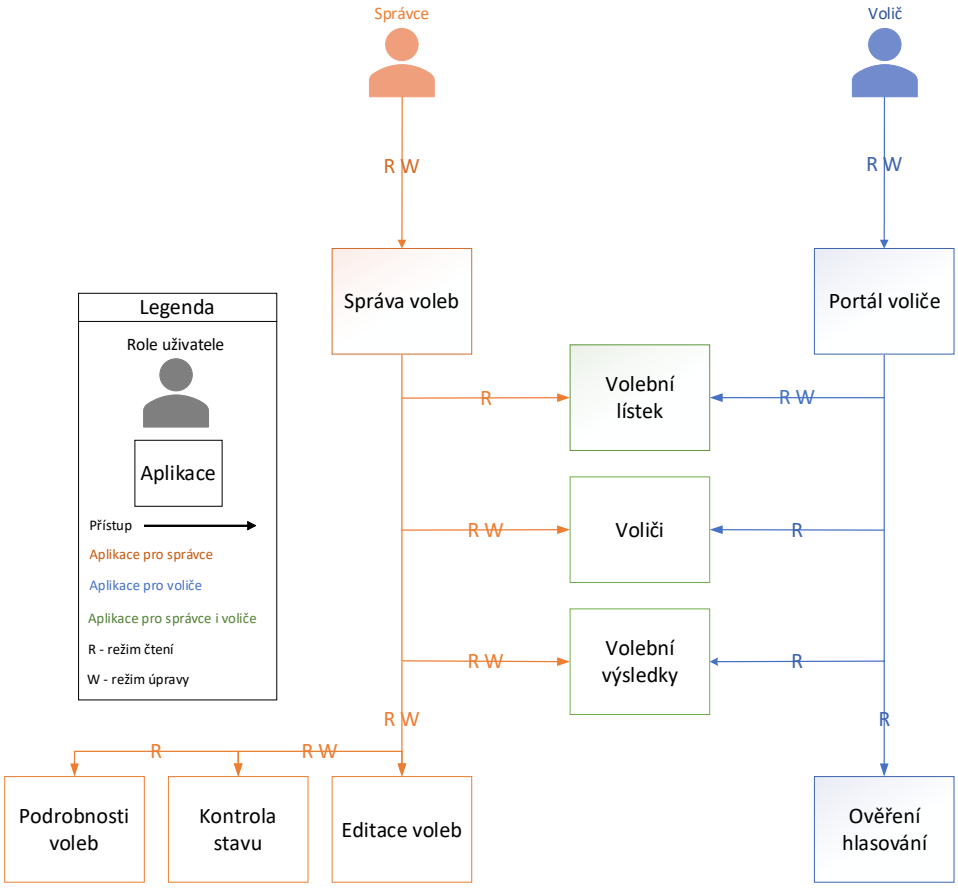
3.4.6 NÁVRH APLIKACE

V následující kapitole navrhnu aplikace, které se budou používat pro správu voleb a pro volení.

Rozlišujeme dva typy přístupů k volbám. Prvním typem přístupu je správce. Správce by měl mít možnost založit volby, přepínat jejich stavy, zobrazit náhled volebního lístku a následně tyto volby uzavřít. Proto vytvořím aplikaci Správa voleb. Do ní budou mít přístup jen správci voleb.

Avšak volič nemůže upravovat volby, ani jejich stavy, ale má možnost ve volbách hlasovat. Proto pro něj vytvořím aplikaci Portál voliče. Tuto možnost ale nemá správce (za předpokladu, že není zároveň i volič daných voleb).

Pro rozdělení kódu do samostatných celků navrhuji vytvořit následující aplikace: Správa voleb, Editace voleb, Kontrola stavu, Podrobnosti voleb, Portál voliče, Volební lístek, Voliči, Volební výsledky, Ověření hlasování



Obrázek 13: Mapa stránek

(Zdroj: vlastní zpracování)

V obrázku Mapa stránek je navrhnut přístup dvou hlavních rolí (správce a volič) do aplikací agendy voleb. Z Mapy stránek je názorné, že budou buď aplikace přímo pro správce, přímo pro voliče, nebo se společným přístupem obou rolí.

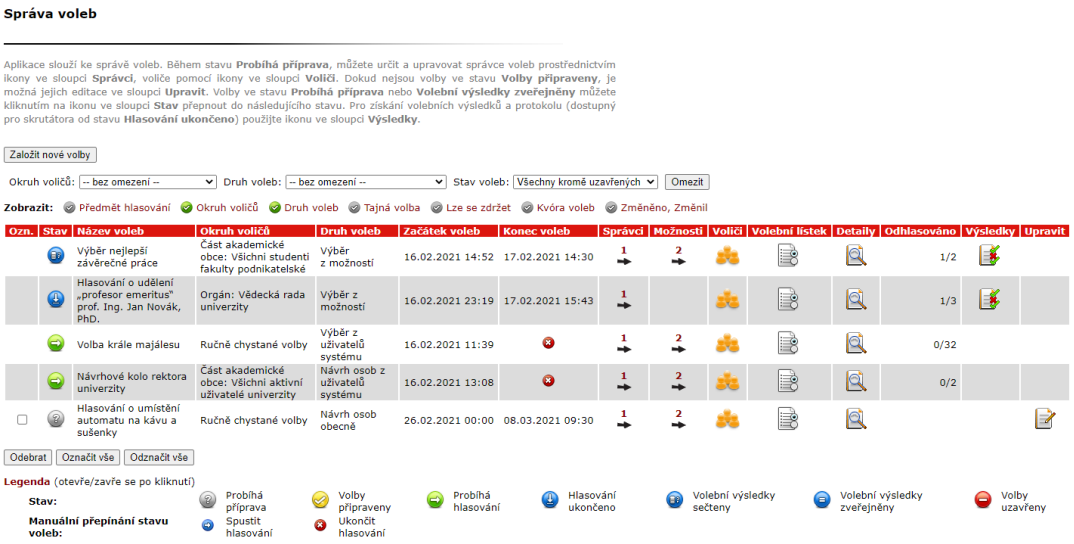
3.5 TVORBA APLIKACE

V následující kapitole popíšu jednotlivé aplikace – představím jejich funkcionalitu, přístupy do aplikace, přístup k datům a vzhled.

3.5.1 SPRÁVA VOLEB

Jako jedna z nejdůležitějších aplikací je Správa voleb. Aplikace poskytuje v základním zobrazení tabulku voleb s jejich detaily o nastavení, stejně tak odkazy do jednotlivých aplikací, které pracují právě s jedněmi volbami.

Aplikace je dostupná pro správce voleb a uživatele s právem „evolby-a“. Práva jsou v UISu atributy voličů, dle kterých se řídí přístup do jednotlivých aplikací. Práva mohou být přiřazena buď ručně (např. práva přístupu studenta do studijních aplikací), nebo nějakou rolí/pravomocí (např. práva děkana). Právo „evolby-a“ bylo vytvořeno pro systémové integrátory, aby mohli vytvářet volby a delegovat toto právo na své kolegy. Aplikace slouží k vytváření voleb, mazání, nastavování, přepínání stavů atd.

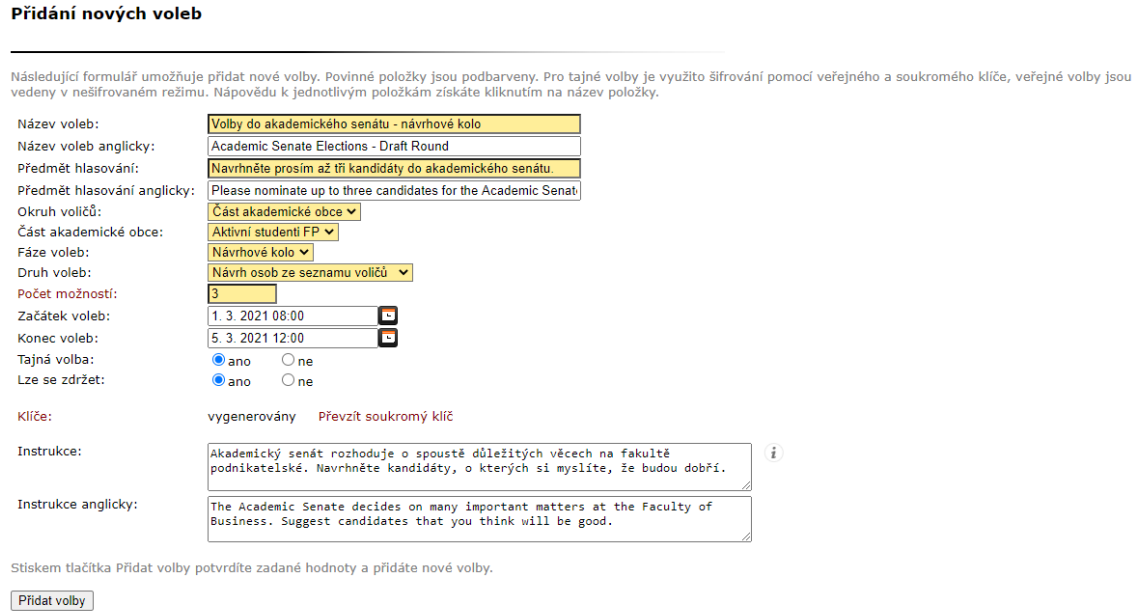


Obrázek 14: Aplikace Správa voleb

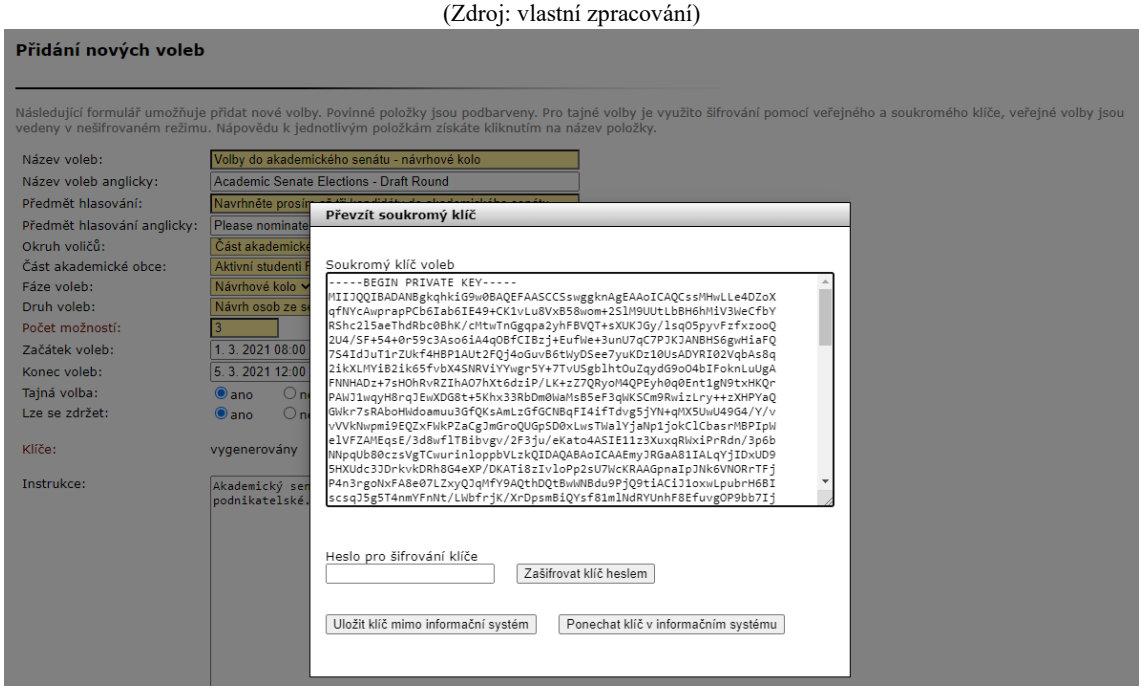
(Zdroj: vlastní zpracování)

V nabídce přidání nových voleb správce voleb vyplní název voleb, předmět voleb, okruh voličů (tzn. definice voličského seznamu) a další nastavení voleb.

Při vytváření voleb je možné stáhnout privátní klíč mimo systém (například pro uložení na USB paměť). Tento klíč je také možné zašifrovat uživatelem zvoleným heslem. Po hlasování – při sčítání hlasů bude správce vyzván k zadání klíče, respektive hesla.



Obrázek 15: Aplikace Správa voleb – Přidání nových voleb
(Zdroj: vlastní zpracování)



Obrázek 16: Aplikace Správa voleb – Převzít soukromý klíč
(Zdroj: vlastní zpracování)

3.5.2 VOLIČI

Aplikace Voliči je převážně určena pro správce pro definování voličského seznamu.

V UISu existují definice různých skupin uživatelů, zjednodušeně by se dalo říct, že se jedná o databázové pohledy. Může se jednat například o pohled „Všichni aktivní studenti akreditovaných studijních oborů“ nebo třeba „Zaměstnanci rektorátu na méně než plný úvazek“… Pohledů je opravdu velké množství, proto byly do voleb filtrovány jen některé – jedná se o orgány a části akademické obce.

Agenda voleb ale nemůže pracovat s pohledy. Představme si, že ve volbách budou volit všichni studenti členy do akademického senátu. V průběhu voleb by se ale počet voličů mohl měnit, což je nepřípustné. Některé hlasy by byly podle aplikace neplatné, některé by naopak platné byly i přes to, že by student v průběhu hlasování mohl přerušit studium atd. Proto je potřeba voličský seznam k určitému stavu zafixovat. Fixace probíhá přepnutím do stavu „Volby připraveny“.

Obrázek 17: Aplikace pro správu voličů – Automatické plnění voličů

(Zdroj: vlastní zpracování)

V průběhu přípravy voleb přece jen může dojít k nerovnosti mezi tabulkou „Volby_volici“ a definicí voličského seznamu. Pro tyto situace slouží záložka „Automatické plnění voličů“, která porovná seznam voličů oproti definici voličského seznamu. Poté případně oznámí změny – kdo přebývá a kdo by měl být přidán.

3.5.3 KONTROLA STAVU

Kontrola stavu je funkcionalita, která se využívá při přepínání stavu voleb z „Probíhá příprava“ do stavu „Volby připraveny“. V této nabídce se zkontroluje, zda volby jsou správně nastaveny, konkrétně:

- Volby mají alespoň jednoho správce
- Volby mají vyplněny možnosti (tzn. je z čeho vybírat)
- Voličský seznam je aktuální

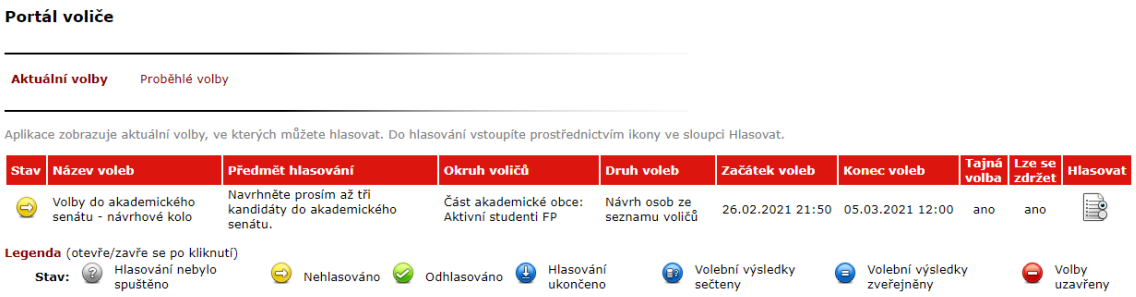
Po splnění těchto podmínek je možné volby potvrdit a přepnout do stavu „Volby připraveny“.

Obrázek 18: Správa voleb – Kontrola stavu

(Zdroj: vlastní zpracování)

3.5.4 PORTÁL VOLIČE

Aplikace Portál voliče slouží jako přehledová nabídka pro voliče. V záložce Aktuální volby může vybrat volby, ve kterých se chce zúčastnit hlasování. V záložce Proběhlé volby může prohlížet výsledky voleb nebo ověřit své hlasování.

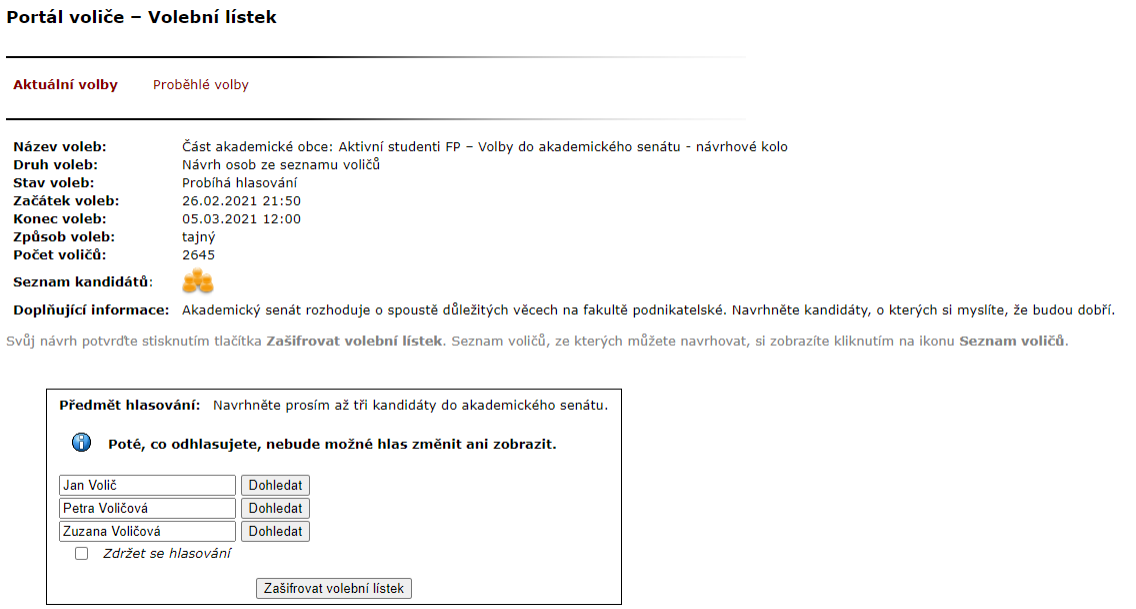


Obrázek 19: Aplikace Portál voliče

(Zdroj: vlastní zpracování)

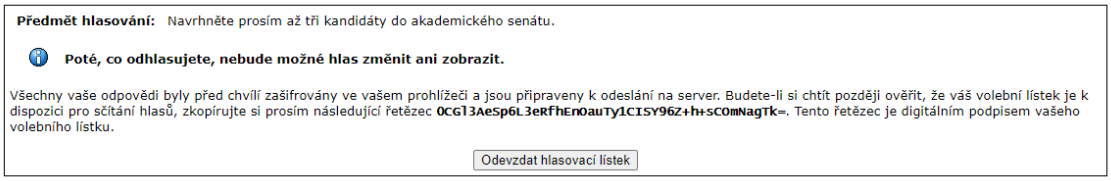
3.5.5 VOLEBNÍ LÍSTEK

Aplikace Volební lístek slouží k hlasování. Volič zaškrtne, vybere či vypíše (dle typu voleb) svůj hlas a stiskne na tlačítko Zašifrovat volební lístek. Po stisku tlačítka Javascriptová funkce zanalyzuje parametry volebního lístku a zašifruje je. Pouze zašifrovaná data spolu s informací, kdo volil a v jakých volbách, se pošlou zpátky na server, kde jsou uložena zašifrovaně až do doby, než skrutátor dešifruje a sečte volební výsledky.



Obrázek 20: Portál voliče – Volební lístek

(Zdroj: vlastní zpracování)



Obrázek 21: Odeslání zašifrovaného hlasovacího lístku

(Zdroj: vlastní zpracování)

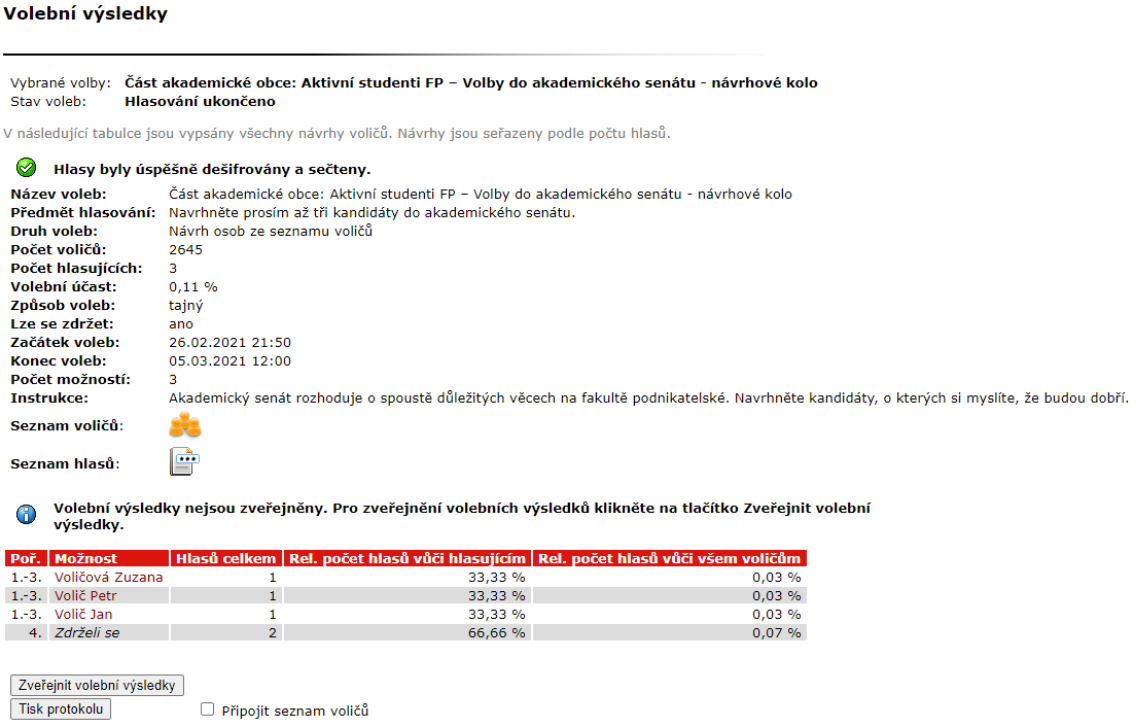
3.5.6 VOLEBNÍ VÝSLEDKY

Po ukončení hlasování je skrutátor informován a je systémem vyzván k Dešifrování a sečtení volebních hlasů. Za předpokladu, že privátní klíč není uložený v systému nebo je zaheslovaný, skrutátor pomocí dialogového okna vloží klíč, respektive heslo.

Po kontrole privátního klíče jsou pomocí Javascriptové funkce v prohlížeči data dešifrována a připravena k sečtení – uložení v čisté podobě do databáze. Poté jsou skrutátorovi zobrazeny výsledky. Skrutátor po revizi výsledků a tisku protokolu zveřejní volební výsledky voličům. Ti pak budou mít do výsledků přístup přes již zmiňovaný Portál voliče.

Obrázek 22: Volební výsledky před započtením a zveřejněním hlasů

(Zdroj: vlastní zpracování)



Volební výsledky

Vybrané volby: **Část akademické obce: Aktivní studenti FP – Volby do akademického senátu - návrhové kolo**
Stav voleb: **Hlasování ukončeno**

V následující tabulce jsou vypsány všechny návrhy voličů. Návrhy jsou seřazeny podle počtu hlasů.

Hlasy byly úspěšně dešifrovány a sečteny.

Název voleb:	Část akademické obce: Aktivní studenti FP – Volby do akademického senátu - návrhové kolo
Předmět hlasování:	Navrhněte prosím až tři kandidáty do akademického senátu.
Druh voleb:	Návrh osob ze seznamu voličů
Počet voličů:	2645
Počet hlasujících:	3
Volební účast:	0,11 %
Způsob voleb:	tajný
Lze se zdržet:	ano
Začátek voleb:	26.02.2021 21:50
Konec voleb:	05.03.2021 12:00
Počet možností:	3
Instrukce:	Akademický senát rozhoduje o spoustě důležitých věcech na fakultě podnikatelské. Navrhněte kandidáty, o kterých si myslíte, že budou dobří.
Seznam voličů:	
Seznam hlasů:	

Volební výsledky nejsou zveřejněny. Pro zveřejnění volebních výsledků klikněte na tlačítko Zveřejnit volební výsledky.

Poř.	Možnost	Hlasů celkem	Rel. počet hlasů vůči hlasujícím	Rel. počet hlasů vůči všem voličům
1.-3.	Voličová Zuzana	1	33,33 %	0,03 %
1.-3.	Volič Petr	1	33,33 %	0,03 %
1.-3.	Volič Jan	1	33,33 %	0,03 %
4.	*Zdrželi se*	2	66,66 %	0,07 %

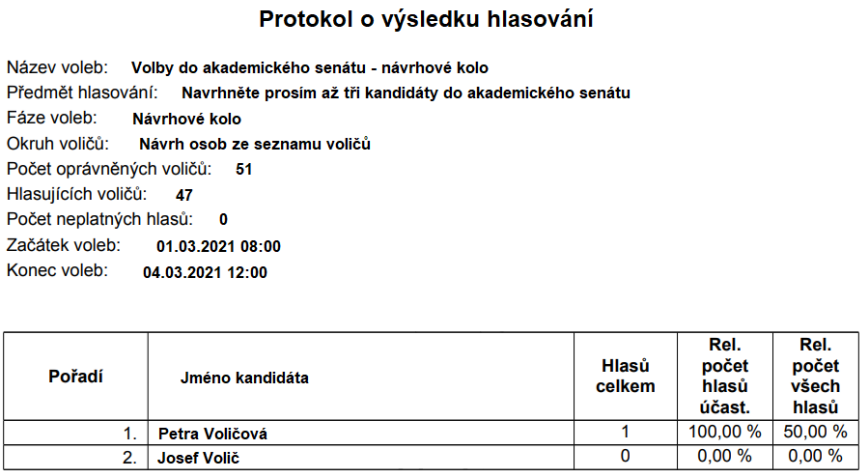
Zveřejnit volební výsledky
Tisk protokolu
Připojit seznam voličů

Obrázek 23: Volební výsledky po započtení hlasů

(Zdroj: vlastní zpracování)

3.5.7 PROTOKOL O VÝSLEDKU HLASOVÁNÍ

Protokol o výsledku hlasování je nedílnou součástí všech voleb – je to průkazný dokument o výsledku hlasování na akademické půdě a obsahuje všechny nutné informace k identifikaci voleb a jejich výsledku. Tento dokument je archivován. Při tisku je volitelné připojit k dokumentu i seznam voličů daných voleb.



Protokol o výsledku hlasování

Název voleb: **Volby do akademického senátu - návrhové kolo**
Předmět hlasování: **Navrhněte prosím až tři kandidáty do akademického senátu**
Fáze voleb: **Návrhové kolo**
Okruh voličů: **Návrh osob ze seznamu voličů**
Počet oprávněných voličů: **51**
Hlasujících voličů: **47**
Počet neplatných hlasů: **0**
Začátek voleb: **01.03.2021 08:00**
Konec voleb: **04.03.2021 12:00**

Pořadí	Jméno kandidáta	Hlasů celkem	Rel. počet hlasů účast.	Rel. počet všech hlasů
1.	**Petra Voličová**	1	100,00 %	50,00 %
2.	**Josef Volič**	0	0,00 %	0,00 %

Obrázek 24: Protokol o výsledku hlasování

(Zdroj: vlastní zpracování)

3.5.8 SHRNUTÍ TVORBY APLIKACÍ

Zmíněné aplikace by měly pokrýt celou základní škálu funkcí agendy e-voleb. Není však vyloučené, že se počet aplikací rozšíří například jako rozšíření nějaké již existující agendy s automatizovaným zakládáním voleb (toto je již ve vývoji pro agendu státních závěrečných zkoušek, z rozsahu a zadání této práce se však tomuto rozšíření věnuji jen okrajově).

3.6 TESTOVÁNÍ APLIKACE

Poté, co je vytvořena aplikace, přichází na řadu testování aplikace.

Oddělení QA (Quality Assurance) má na starost testování aplikace, než je předána zákazníkovi. Po vytvoření jednotlivých aplikací zpřístupním aplikace na vývojovém serveru a předám instrukce testerům. Proces interního testování se odehrává v systému Redmine. V požadavku, který vytvořím, je v hlavičce informace, jaké soubory se testují. Posléze jsou v bodech popsány aplikace – jak se mají chovat, s jakými právy kterým uživatelům zobrazují data apod. Testeři mají právo využívat tzv. „cizí identitu“ – parametr obsahující identifikační číslo uživatele, takže je možné s přesností zjistit, jaký obsah je kterým uživatelům zobrazen. Testeři mají za úkol nasimulovat chování uživatele, celou aplikaci projít a zjistit případné nedostatky. Pokud jsou nalezeny, je dotyčný vývojář systémem upozorněn na vrácení – po úpravě požadavek znovu přepne na oddělení QA.

3.7 ZVEŘEJNĚNÍ APLIKACE

Poté, co již nejsou nalezeny žádné nedostatky, je možné zveřejnit kód zákazníkovi. Pokud se jedná o opravu již stávající aplikace či drobnou změnu, je vynechán krok testování zákazníkem. V případě nové agendy elektronického hlasování je testování nezbytné.

3.7.1 TESTOVÁNÍ APLIKACE ZÁKAZNÍKEM

Jedná se o zveřejnění aplikací pouze na testovací server zákazníka. Každý zákazník má dvě instalace – produkční a testovací.[6]

Systémoví integrátoři si tudíž mohou vyzkoušet funkcionalitu „nanečisto“ – využívat „cizí identity“ pro hlasování a seznámit se s aplikací jako takovou. Po týdnu se data na testovacím serveru přepíšou aktuální kopií z produkčního serveru, proto systémoví integrátoři mohou cvičně pracovat s relativně aktuálními daty.

3.7.2 ŠKOLENÍ SYSTÉMOVÝCH INTEGRÁTORŮ

U vydání nových agend je běžné pořádat školení systémových integrátorů. Školení probíhá videokonferenčně s více zákazníky zároveň.

Termín školení je naplánován na 11. června 2021. Harmonogram školení je zachycen v následující tabulce.

Tabulka 17: Harmonogram školení systémových integrátorů

(Zdroj: vlastní zpracování)

Čas	Téma	Vede
13:30 – 14:30	Slovo úvodem, představení aplikací e-voleb	QA, vývojář e-voleb
14:30 – 15:00	Prostor pro dotazy, zakončení	QA, vývojář e-voleb

Po úvodním slovu vedoucí QA představí aplikace e-voleb. Pomocí sdílené obrazovky počítače vytvoří volby, spustí je a postupně projde všechny kroky a upozorní na důležité detaily z pozice voliče i správce. Posléze je vyhrazen prostor na dotazy.

[6] Firma UIS má ještě vývojovou instalaci, která je přístupná jen zaměstnancům společnosti

3.7.3 SEPSÁNÍ DOKUMENTACE

Vedlejší činností oddělení QA je i sepsání dokumentace zákazníkům. Jedná se o podrobný návod dostupný v konkrétní aplikaci. Dokumentace oddělení QA sepisuje v součinnosti s vývojářem, který aplikaci vyvinul.

3.8 POPROJEKTOVÁ FÁZE

Poprojektová fáze nastává vydáním aplikace zákazníkovi. Během ní se hodnotí práce na aplikaci, její přínosy a řeší se finanční plnění se zákazníkem.

3.8.1 OSTRÝ PROVOZ APLIKACE

Jakmile jsou systémoví integrátoři jednotlivých vysokých škol proškolení, tak obvykle do několika dní požádají vývojáře, aby agendu spustili i na produkční instalaci.

Jednotlivé spouštění a vypínání funkcionalit probíhá na master serveru. Aplikace na správu funkcionalit se nazývá „Operativní řízení na instalacích“. Jelikož všechny instalace obsahují stejný software i datové schéma, tak je zapínání funkcionalit řešeno pomocí databázové tabulky, která je po každé změně na Master serveru rozeslána na všechny instalace.

Pro zapnutí e-voleb na jednotlivých produkčních instalacích stačí přepsat logickou hodnotu u jednotlivých instalací a tuto tabulku nechat synchronizovat. Po provedení těchto změn jsou informovaní systémoví integrátoři.

3.8.2 FINANČNÍ ZHODNOCENÍ PROJEKTU

Odhad celkové ceny vychází z časové analýzy a předpokladu, že termín a náročnost byla přesně odhadnuta. Jelikož projekt v době vydání diplomové práce stále běží, není možné počítat s konečnými vynaloženými časovými náklady. V této kapitole je čas vykazován v hodinách, respektive člověkohodinách.

Vzhledem k tomu, že ceník je součástí obchodního tajemství IS4U, kalkulace hodinových sazeb jsou průměrnými sazbami z výzkumu zpracovaným na Vysoké škole ekonomické analyzující smlouvy registru smluv České republiky mezi roky 2016 a 2018. [36]

Mezi nejnáročnější činnosti v celém projektu je samotná tvorba aplikací elektronického hlasování. Jen samotná tvorba podle analýzy PERT zabere přibližně 265 hodin, což je cca 50 % celkového objemu práce. Testovací provoz aplikace bude trvat přibližně 7 pracovních člověkodní, během kterých nebude nutná kooperace ze strany IS4U. V následující tabulce je vyčíslena náročnost podle jednotlivých pozic v rámci projektu. [36]

Tabulka 18: Práce na jednotlivých úkolech

(Zdroj: vlastní zpracování)

ID	Popis činnosti	Programátoři	Vedoucí projektu	Databázový architekt	Testeři	Školitel	t[ij] [hod]
A	Návrh elektronického hlasování v UIS	31,3	0	0	0	0	31,3
B	Design Review	5	10,3	0	0	0	15,3
C	Výběr a volba technologie	8	0	0	0	0	8,0
D	Schůze se zákazníky k diskusi nad návrhem	4	4	0	0	0	8,0
E	Schůze s kolegy a vedením	2	2	1	2	1	8,0
F	Návrh datové struktury	10	0	12	0	0	22,0
G	Návrh backendu a frontendu	24	7,3	0	0	0	31,3
H	Tvorba aplikací pro správce	100	14,7	0	0	0	114,7
I	Code Review aplikací pro správce	5	9	0	0	0	14,0
J	Testování aplikací pro správce	10	3	0	18,3	0	31,3
K	Schůze se zákazníky k diskusi nad průběhem	4	2	0	1,3	0	7,3
L	Zveřejnění aplikací pro správce	3	2	1	0	0	6,0
M	Tvorba aplikací pro voliče	130	19,3	0	0	0	149,3
N	Code Review aplikací pro voliče	6	12,7	0	0	0	18,7
O	Testování aplikací pro voliče	6	16,7	0	0	0	22,7
P	Zveřejnění aplikací pro voliče na testování	4	1,8	1,5	0	0	7,3
Q	Školení zákazníků	4,7	0	0	0	8	12,7
R	Testovací provoz (testování zákazníkem)	0	0	0	0	0	55,3
S	Sepsání dokumentace	3	0	0	0	16,3	19,3
T	Poprojektová fáze	8	14	1	1	4	28,0

V tabulce 19 je zahrnuta průměrná hodinová sazba, počet hodin strávených na projektu a cena. Celková cena projektu včetně návrhu, tvorby a implementace byla vyčíslena na **717 589 Kč bez DPH**.

Tabulka 19: Cena za provedení projektu

(Zdroj: vlastní zpracování, [36])

Pozice	Průměrná hod. sazba	Počet hod.	Cena
Programátoři	1 318 Kč	368	484 886 Kč
Vedoucí projektu	1 556 Kč	118,8	184 868 Kč
Testeři	1 157 Kč	22,6	26 140 Kč
Databázový architekt	1 315 Kč	16,5	21 695 Kč
Školitel	1 293 Kč	29,3	37 896 Kč
Celkem	_	**525,9**	**717 589 Kč**

Celkové vyčíslení a fakturace spadá do poprojektové fáze. V ní je také vhodné zpětně zhodnotit odhady učiněné v PERT analýze. Tím je možné neustále zlepšovat odhady, a tudíž se vyvarovat nečekaným nákladům. Zpětné hodnocení je mimo jiné důležitým prvkem v agilním systému řízení, které pro tento projekt nebyl využit.

3.8.3 NEFINANČNÍ PŘÍNOSY PROJEKTU

Z toho důvodu, že nabízená cena za produkt není veřejná, finančním zhodnocením projektu se nadále nebudu zabývat.

Níže nastíním hlavní přínosy aplikace pro elektronické hlasování.

Výhoda on-line hlasování

Přístup k hlasování v době pandemie [37] může být jednodušší online než fyzická přítomnost hlasujících. Přesunutím hlasování do online prostoru odpadá nutnost fyzické přítomnosti na zasedání/volbách, a tudíž otevírá možnosti pro hlasování odkudkoliv na světě – z domácí pracovny, izolace, karantény, zahraniční cesty či dovolené.

Snížení organizační náročnosti

Online volby snižují organizační náročnost, a to zejména pro volby o mnoha voličích. Například při volbách do akademického senátu odpadá organizační náročnost – nemusí být vytvořena volební komise, odpadá tisk a manuální kontrola hlasovacích lístků, nebo kontrola studentských či učitelských průkazů.

Využití při státnicích

Jako další přinos aplikace e-voleb je možnost využití této agendy pro státní závěrečné zkoušky studentů. Mimo popisované schéma je v plánu vytvořit aplikaci na zakládání voleb pro státní závěrečné zkoušky. Tyto volby může komise využít, například pokud se

škola vydala cestou distančního skládání státních závěrečných zkoušek pomocí videokonference, jak to například udělala již druhým rokem brněnská Fakulta sociálních studií Masarykovy univerzity, katedra mediálních studií a žurnalistiky. [38]

Průkaznost

Za předpokladu dobrého návrhu hlasování v UIS, aniž by při tvorbě došlo ke kritickým chybám, tak je průkaznost těchto voleb relativně silná.

Spekulace ze strany voličů mohou nastat vždy, avšak pro podporu důvěryhodnosti této aplikace je nezávazně domluveno s jednou nejmenovanou vysokou školou, že pořídí nezávislý audit této aplikace. Pokud audit potvrdí, že je návrh, tvorba i implementace aplikace bezchybná, zvýší se důvěra mezi voliči napříč všemi vysokými školami, jelikož je kód a databázové schéma na všech instalacích stejný.

3.9 MOŽNÉ ROZŠÍŘENÍ DO BUDOUCNA

Aplikace poskytuje všechny základní funkce, které by měl systém elektronického hlasování nabízet. Aplikace splňují všechny požadavky zákazníka. Je však možné, že v průběhu používání si jednotlivé vysoké školy budou přát nějakou funkcionalitu rozšířit dle individuálních požadavků. V této kapitole nastíním možná rozšíření do budoucna, která vyvstala v průběhu návrhu a již se do základního balíku aplikací nevešla.

3.9.1 PRŮBĚŽNÉ PODEPISOVANÍ VÝSLEDKŮ NEZÁVISLÝMI AUTORITAMI

Pro zvýšení důvěryhodnosti volebních výsledků je v plánu doplnit v další fázi vývoje e-voleb průběžné podepisování výsledků nezávislými autoritami. Každá vysoká škola by měla server pro podepsání jednotlivých hlasů. Pokud by na vysoké škole A probíhaly volby, každý odeslaný hlasovací lístek by byl podepsán veřejným klíčem školy A. Toto podepisování by však proběhlo na všech školách, tedy nejen školou A, ale i školou B a školou C. Podpis by se na všech třech školách uschoval a při dešifrování a sčítání voleb na škole A by byly ostatní školy požádány k předložení všech podpisů.

Pokud by podpisy nebyly na všech školách stejné, došlo by k nekonzistentnosti, nebo podvodu a výsledek voleb by mohl být napaden. Pokud se budou všechny podpisy rovnat, celková důvěryhodnost výsledků je výrazněji posílena než bez této funkcionality.

Časové razítko protokolu a výsledků voleb

Jako další možné rozšíření, které by podpořilo důvěrnost ve volební výsledky, by bylo opatření digitálního razítka při vytváření volebního protokolu a přidružení k dokumentu. Autorit vydávající digitální razítka je v Česku a na Slovensku několik. Jsou to například Česká pošta nebo Ministerstvo obrany Slovenské republiky. Služba je však zpoplatněna. [39]

Výše popsaná možná rozšíření do budoucna nejsou však vyčerpávající. Další rozšíření se bude vyvíjet na základě zákazníkova používání aplikace pro elektronické hlasování.

ZÁVĚR

Cílem diplomové práce bylo zanalyzovat, navrhnout a implementovat webovou aplikaci do prostředí firmy.

V první části práce jsem se věnoval teoretickým základům z oblasti informačních systémů, webových aplikací a kybernetiky. Popsal jsem pojmy jako informace, proces, data, informační systém apod. Kapitolu jsem doplnil o výklady pojmů jako riziková a časová analýza, volby a hlasování, se kterými jsem operoval v analytické a návrhové části.

Ve druhé části práce jsem představil firmu IS4U, pro kterou byla analýza prováděna. Popsal jsem stručně její historii, organizační strukturu, produkt a zákazníky, na které byl návrh zaměřen. Zmapoval jsem její procesy, informační systém a provedl jsem dotazníkové šetření se zákazníky. Ti mi popsali požadavky na aplikaci elektronického hlasování. Jejich požadavky jsem shrnul do jednotlivých kritérií, podle kterých jsem prvně zmapoval trh s již existujícími řešeními. Jelikož všechna navrhovaná řešení dramaticky nesplňovala požadavky definované zákazníky, vedení firmy se s nimi domluvilo na provedení návrhu, vytvoření a implementaci nové webové aplikace v rámci informačního systému.

Třetí část začíná rizikovou a časovou analýzou procesu návrhu, tvoření a implementace aplikace. Poté je předložen návrh na vytvoření webové aplikace. Tento návrh je zpracován jak z hlediska EPC diagramu a RACI matice, tak je podrobně popsáno chování navrhované aplikace. Při návrhu byl kladen důraz na splnění všech vytyčených kritérií v analytické části. Po navržení databázového schématu je popsána tvorba jednotlivých aplikacích a jejich funkcionalit. K aplikacím jsou přidány screenshoty pro doplnění představy.

Po návrhu a praktickém vytvoření jsou popsány kroky implementace s návrhem školení zákazníků. Celou práci uzavírám vyčíslením přibližných nákladů a shrnutím přínosů nově vytvořené webové aplikace.

Všechny stanovené cíle byly splněny.

SEZNAM POUŽITÉ LITERATURY

[1] MOLNÁR, Zdeněk. *Efektivnost informačních systémů*. 2. rozš. vyd. Praha: Grada, 2001. Management v informační společnosti. ISBN 80-247-0087-5.

[2] GÁLA, Libor, Jan POUR a Prokop TOMAN. *Podniková informatika: počítačové aplikace v podnikové a mezipodnikové praxi, technologie informačních systémů, řízení a rozvoj podnikové informatiky*. 1. vyd. Praha: Grada, 2006. Management v informační společnosti. ISBN 80-247-1278-4.

[3] GÁLA, Libor, Jan POUR a Zuzana ŠEDIVÁ. *Podniková informatika*. 2., přeprac. a aktualiz. vyd. Praha: Grada, 2009. Expert (Grada). ISBN 978-80-247-2615-1.

[4] HARDCASTLE, E. *Business Information Systems*. Ventus Publishing ApS, 2008. ISBN 978-87-7681-463-2.

[5] MOLNÁR, Zdeněk. *Moderní metody řízení informačních systémů*. V Praze: Grada, 1992. Nestůjte za dveřmi (Grada). ISBN 80-856-2307-2.

[6] KOCH, Miloš a Bernard NEUWIRTH. *Datové a funkční modelování*. Vyd. 4., rozš. Brno: Akademické nakladatelství CERM, 2010. ISBN 978-80-214-4125-5.

[7] A Relational Database Overview. *Oracle Java Documentation: The Java™ Tutorials* [online]. Redwood City, CA, USA: Oracle, 2020 [cit. 2020-12-19]. Dostupné z: https://docs.oracle.com/javase/tutorial/jdbc/overview/database.html

[8] SODOMKA, Petr a Hana KLČOVÁ. *Informační systémy v podnikové praxi*. 2., aktualiz. a rozš. vyd. Brno: Computer Press, 2010. ISBN 978-80-251-2878-7.

[9] OULEHLA, Milan a Roman JAŠEK. *Moderní kryptografie*. [Praha]: IFP Publishing, 2017. ISBN 978-80-87383-67-4.

[10] LEEUWEN, J.V. *Handbook of theoretical computer science*. Vyd. 1. London: Elesevier, 1990. ISBN 0-444-88074-7.

[11] ČERMÁK, Miroslav. Autentizace. *Clever and Smart* [online]. Dolní Břežany: Miroslav Čermák, 2009 [cit. 2020-12-22]. Dostupné z: https://www.cleverandsmart.cz/autentizace/

[12] MALINKA, Kamil. *Kryptografie a informační bezpečnost: Základy kryptologie*. Brno: VUT, 2008.

[13] DOSTÁLEK, Libor. *Velký průvodce protokoly TCP/IP: bezpečnost*. 2. aktualiz. vyd. Praha: Computer Press, 2003. ISBN 80-722-6849-X.

[14] VONDRÁK, Ivo. *Průvodce kurzem Metody byznys modelování: eLearning : distanční forma studia : vzdělávací řídící systém MOODLE*. 1. vyd. Ostrava: VŠB - Technická univerzita, Centrum eLearningu VIRTUNIV, 2004. ISBN 80-248-0729-7.

[15] RACI matice: Struktura pro zobrazení vztahů mezi balíky práce a členy týmu, která napomáhá zajistit, aby každá část projektových prací byla někomu přiřazena a byla jasná odpovědnost. *Projectman: connecting experts* [online]. Praha: eBRÁNA, Projectman.cz [cit. 2021-04-30]. Dostupné z: https://www.projectman.cz/sablony/raci-matice

[16] SCAMBRAY, Joel a Mike SHEMA. *Hacking bez tajemství: Webové aplikace*. 1. vyd. Brno: Computer Press, 2003. ISBN 80-7226-769-8.

[17] PRETTYMAN, S. *Learn PHP 7: object oriented modular programming using HTML5, CSS3, Javascript, XML, JSON, and MYSQL* [online]. Apress, 2015 [cit. 2021-03-17]. ISBN 978-1-4842-1730-6.

[18] ECMAScript® 2021 Language Specification. *TC39: Specifying JavaScript*. [online]. Geneva (Switzerland): ecma, 2020 [cit. 2020-12-15]. Dostupné z: https://tc39.es/ecma262/

[19] Perl Documentation: perlhist. *PerlDoc Browser* [online]. Los Angeles: Larry Wall, 2020 [cit. 2020-12-15]. Dostupné z: https://perldoc.perl.org/perlhist

[20] The Fall Of Perl, The Web's Most Promising Language: And the rise of Python. Does Perl have a future?. *Fast Company* [online]. New York (U.S.): Fast Company,

2014 [cit. 2020-12-15]. Dostupné z: https://www.fastcompany.com/3026446/the-fall-of-perl-the-webs-most-promising-language

[21] The Common Gateway Interface (CGI) Version 1.1. *IETF Tools: IETF-related tools, standalone or hosted on tools.ietf.org.* [online]. Fremont California (U.S.): IETF, 2020 [cit. 2020-12-15]. Dostupné z: http://tools.ietf.org/html/rfc3875

[22] Časová analýza: Veřejná knihovna Mendelu. *Elektronické studijní materiály* [online]. Brno: Mendelu, 2021 [cit. 2021-04-17]. Dostupné z: https://is.mendelu.cz/eknihovna/opory/zobraz_cast.pl?cast=72594

[23] SMEJKAL, Vladimír a Karel RAIS. *Řízení rizik ve firmách a jiných organizacích*. 4., aktualiz. a rozš. vyd. Praha: Grada, 2013. Expert (Grada). ISBN 978-80-247-4644-9.

[24] HEYWOOD, Andrew. *Politics*. GB: Macmillan, 1997. ISBN 9780333645109.

[25] *Robert's Rules of Order: Pocket Manual of Rules Of Order For Deliberative Assemblies*. U.S.: BiblioBazaar, 2004. ISBN 978-1434604842.

[26] *IS4U s.r.o.* [online]. Brno: IS4U s.r.o., 2021 [cit. 2021-01-26]. Dostupné z: https://www.is4u.cz/cs/index

[27] Úplný výpis z obchodního rejstříku: IS4U, s.r.o., C 65487 vedená u Krajského soudu v Brně. *Veřejný rejstřík a sbírka listin: eJustice* [online]. Praha: Ministerstvo spravedlnosti České republiky, 2015 [cit. 2021-03-12]. Dostupné z: https://or.justice.cz/ias/ui/rejstrik-firma.vysledky?subjektId=637218&typ=UPLNY

[28] NETREFOVÁ, H. *Ústní sdělení*. Brno, 2020.

[29] Orgány univerzity. *Univerzita Pardubice* [online]. Pardubice: UPCE, 2021 [cit. 2021-04-14]. Dostupné z: https://www.upce.cz/univerzita/organy.html

[30] INFORMACE K VYHLÁŠENÍ NOUZOVÉHO STAVU V ČR. *MŠMT: Ministerstvo školství, mládeže a tělovýchovy* [online]. Praha: MŠMT, 2020 [cit. 2021-01-26]. Dostupné z: https://www.msmt.cz/

[31] *Helios Voting: Documentation site for the Helios Voting System.* [online]. Cambridge (Massachuttes): Ben Adida, 2021 [cit. 2021-04-15]. Dostupné z: https://documentation.heliosvoting.org/home

[32] ARMKNECHT, Frederik, Colin BOYD, Christopher CARR, Kristian GJøSTEEN, Angela JÄSCHKE, Christian REUTER a Martin STRAND. A Guide to Fully Homomorphic Encryption. *Cryptology ePrint Archive* [online]. San Diego, U.S.: UCSD, 2021 [cit. 2021-04-15]. Dostupné z: https://eprint.iacr.org/2015/1192

[33] Pricing: Compare our voting software and service offerings. *EBallot: Voting Software and Services that Generate Serious Impact* [online]. Washington U.S.: Votenet Solutions, Inc., 2021 [cit. 2021-04-14]. Dostupné z: https://www.eballot.com/pricing

[34] LEVINSON, Meridith a Irena KRUPIČKOVÁ. 13 nejčastějších chyb v IT projektech. *CIO: Business World* [online]. Praha: Business World, 2021 [cit. 2021-05-01]. Dostupné z: https://businessworld.cz/business-rizeni-podniku/13-nejcastejsich-chyb-v-it-projektech-6742

[35] *Řídící informační systém IS4U, s.r.o.* [online]. Brno: IS4U, s.r.o., 2021 [cit. 2021-03-05]. Dostupné z: https://master.is4u.cz/?lang=cz

[36] BRUCKNER, , GÁLA, VENCOVSKÝ, HRONEK, KŘÍHA, DVORSKÝ a ŠRUBAŘ. *Přehled obvyklých cen ICT prací: Vysoká škola ekonomická v Praze, Fakulta informatiky a statistiky* [online]. Praha: VŠE, 2018 [cit. 2021-05-14]. Dostupné z: https://www.mvcr.cz/soubor/prehled-cen-ict-praci-201905-pdf.aspx. VŠE.

[37] Koronavirus ve světě: WHO vyhlásila globální pandemii. *Brněnský deník Rovnost* [online]. Brno: Brněnský deník Rovnost, 2020 [cit. 2021-04-18]. Dostupné z: https://brnensky.denik.cz/zpravy-ze-sveta/koronavirus-who-globalni-pandemie-20200311.html

[38] Státní závěrečná zkouška a obhajoba: Aktualizace pro virem stižený JS 2021. *MUNI: Katedra mediálních studií a žurnalistiky* [online]. Brno: MUNI, 2021 [cit.

2021-04-19]. Dostupné z: https://medzur.fss.muni.cz/pro-studenty/magisterske-studium/statni-zaverecna-zkouska/prehled

[39] Důvěryhodný seznam Evropské unie. *Adobe* [online]. Mountain View, (CA) U.S.: Adobe, 2021 [cit. 2021-04-19]. Dostupné z: https://helpx.adobe.com/cz/document-cloud/kb/european-union-trust-lists.html

SEZNAM TABULEK

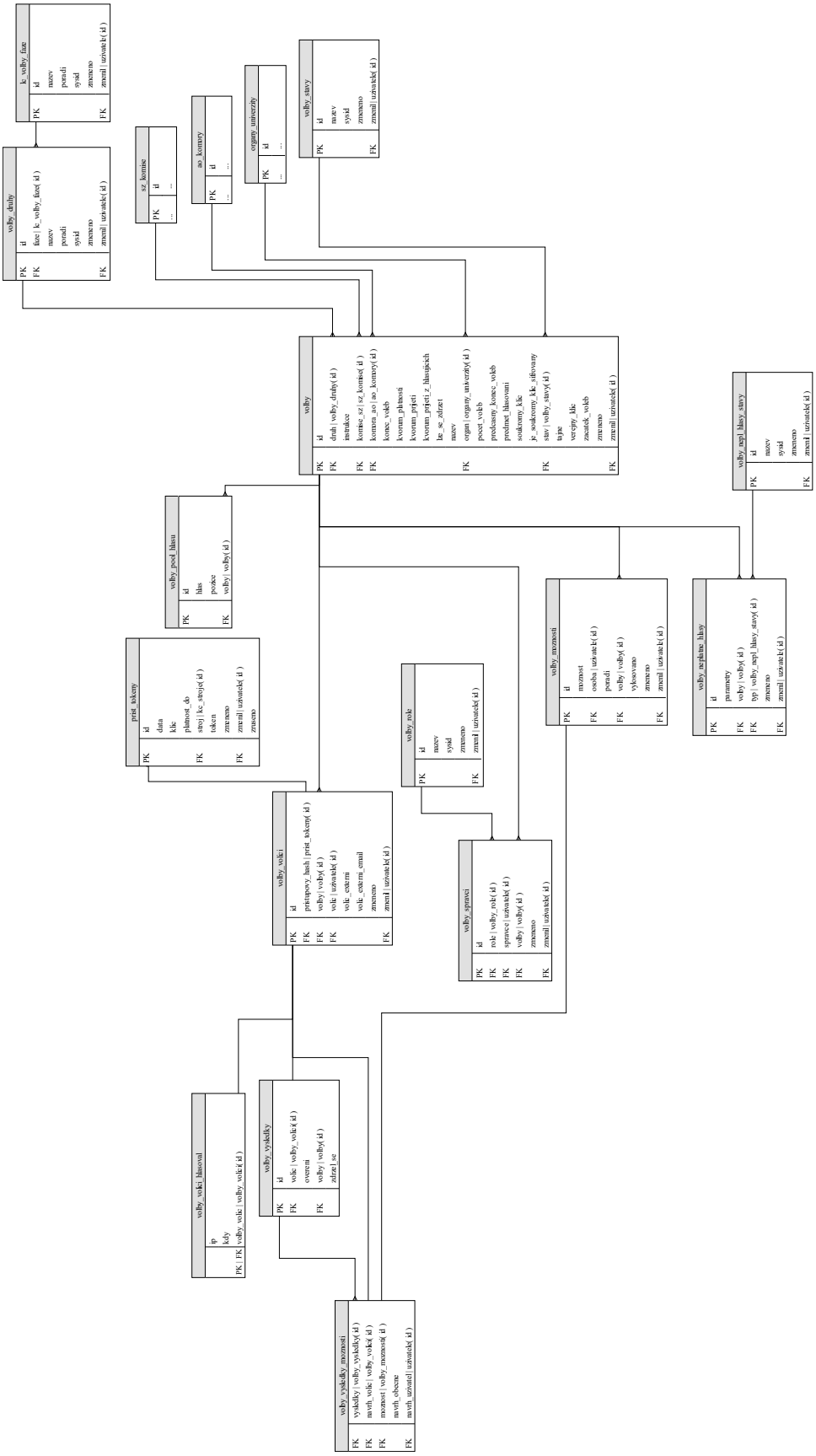
SEZNAM OBRÁZKŮ

PŘÍLOHY

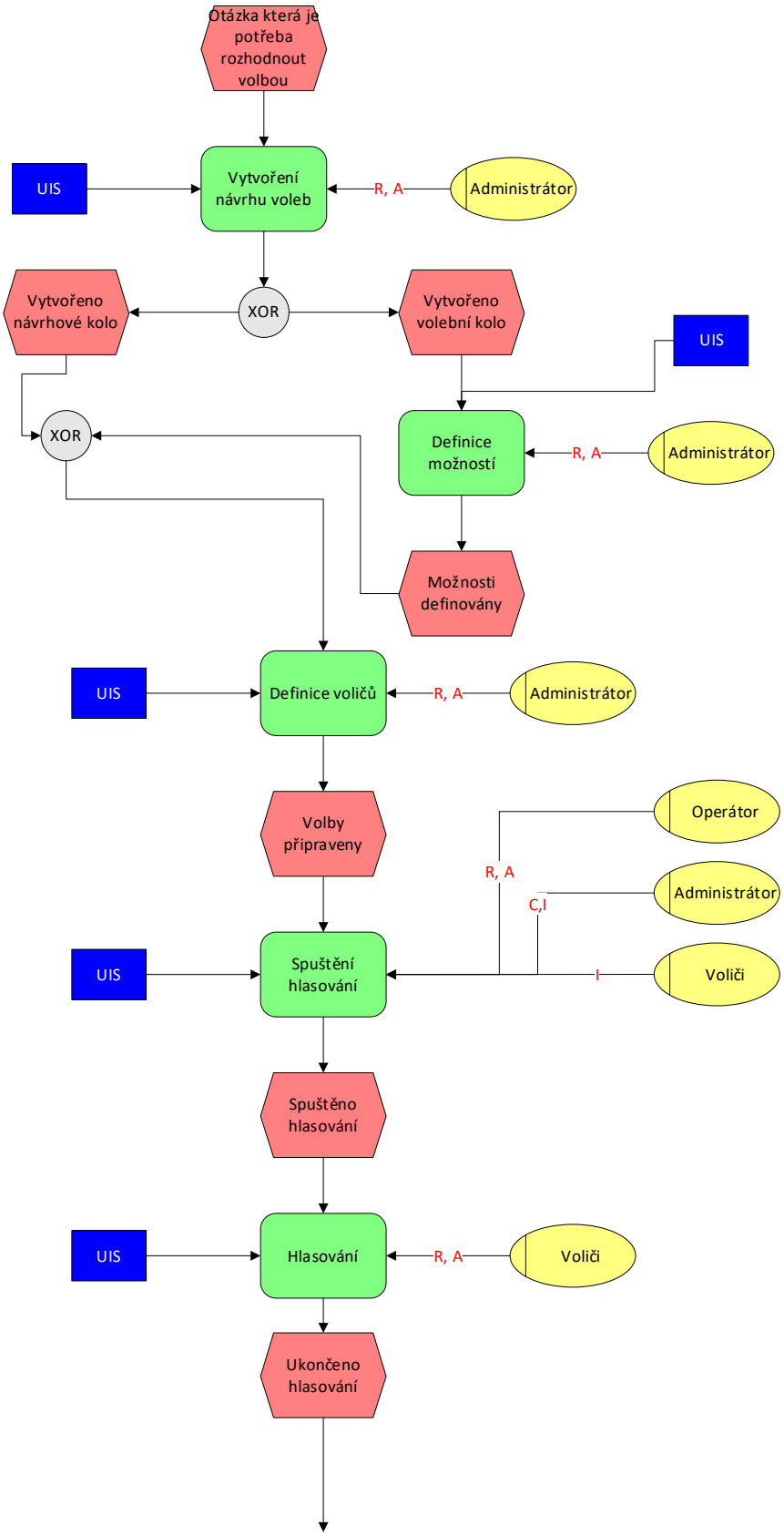
Příloha 1: ERD Diagram

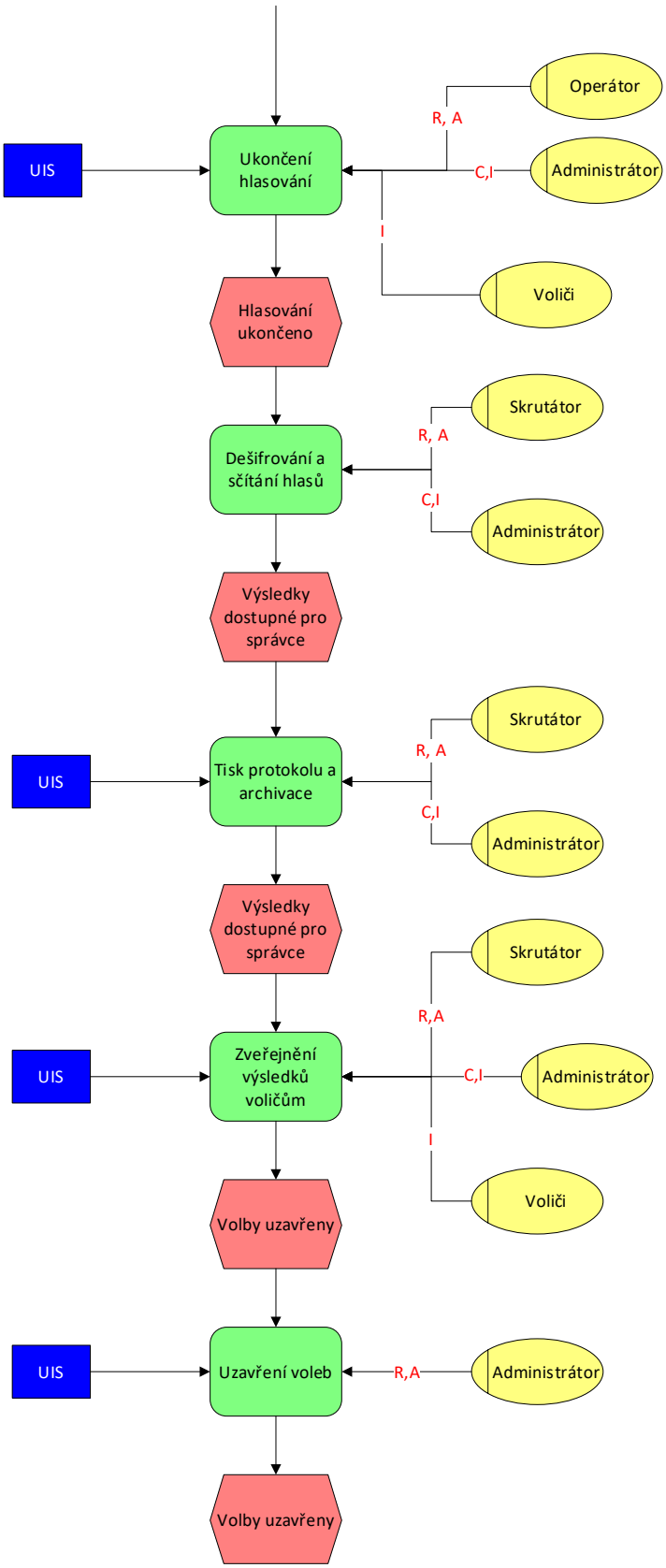
Příloha 2: EPC Diagram voleb s využitím UIS

Příloha 1: ERD Diagram



Příloha 2: EPC Diagram voleb s využitím UIS




Operátor
R, A
UIS
Ukončení hlasování
C,I
Administrátor
I
Voliči
Hlasování ukončeno
Skrutátor
R, A
Dešifrování a sčítání hlasů
C,I
Administrátor
Výsledky dostupné pro správce
Skrutátor
R, A
UIS
Tisk protokolu a archivace
C,I
Administrátor
Výsledky dostupné pro správce
Skrutátor
R,A
UIS
Zveřejnění výsledků voličům
C,I
Administrátor
I
Voliči
Volby uzavřeny
UIS
Uzavření voleb
R,A
Administrátor
Volby uzavřeny